Cisco Transport Manager Release 9.2 GateWay/CORBA User Guide and Programmer Manual

August 23, 2010

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

Text Part Number: OL-20937-01

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco Transport Manager Release 9.2 GateWay/CORBA User Guide and Programmer Manual
© 1999–2010 Cisco Systems, Inc. All rights reserved.

CONTENTS

Preface xiii

Objectives xiii

Audience xiii

Conventions xiv

Related Documentation xiv

Obtaining Documentation and Submitting a Service Request xx

CHAPTER 1 **Overview of CTM GateWay/CORBA** 1-1

1.1 Features of CTM GateWay/CORBA 1-3

1.2 CTM GateWay/CORBA Fault Management 1-3

1.3 Overview of CORBA Services 1-4

1.3.1 Naming Service 1-4

1.3.2 Notification Service 1-4

1.4 Information Model 1-7

1.5 IDL-Related Changes 1-8

1.6 Functionalities Not Supported in R9.2 1-9

1.7 Summary 1-9

CHAPTER 2 **NE- and CTM-Specific Details** 2-1

2.1 NE-Specific Details 2-1

2.1.1 ONS 15216 2-1

2.1.2 ONS 15305 and ONS 15305 CTC 2-2

2.1.3 ONS 15310 CL 2-2

2.1.4 ONS 15310 MA SDH 2-5

2.1.5 ONS 15310 MA SONET 2-6

2.1.6 ONS 15327 2-7

2.1.7 ONS 15454 SDH 2-9

2.1.8 ONS 15454 SONET 2-12

2.1.9 ONS 15530 2-14

2.1.10 ONS 15540 2-16

2.1.11 ONS 15600 SONET 2-18

2.1.12 ONS 15600 SDH 2-21

2.1.13 MGX 8880/8850/8830 2-22

2.1.14 MDS 9000 2-22
2.1.15 Unmanaged NEs 2-22
2.2 CTM-Specific Details 2-23
2.2.1 Layer Rate 2-23
2.2.2 Inventory 2-27
2.2.3 Interfaces 2-28
2.2.4 Object Details 2-29
2.2.5 Naming Conventions 2-30
2.2.6 Programming Details 2-37
2.2.7 Provisioning Subnetwork Connections 2-39

CHAPTER 3 Using CTM GateWay/CORBA Interfaces 3-1
3.1 Overview of APIs Supported by NE Type 3-2
3.2 common::Common_I 3-18
3.2.1 common::Common_I:: getCapabilities 3-18
3.3 emsMgr::EMSMgr_I 3-19
3.3.1 getAllEMSAndMEActiveAlarms 3-19
3.3.2 getAllEMSSystemActiveAlarms 3-20
3.3.3 getAllTopLevelSubnetworks 3-21
3.3.4 getAllTopLevelSubnetworkNames 3-23
3.3.5 getAllTopLevelTopologicalLinks 3-23
3.3.6 getAllTopLevelTopologicalLinkNames 3-25
3.3.7 getTopLevelTopologicalLink 3-26
3.3.8 createTopologicalLink 3-27
3.3.9 createTopologicalLinkWithYCableProtection 3-29
3.3.10 deleteTopologicalLink 3-31
3.3.11 modifyTopologicalLink 3-32
3.3.12 getEMS 3-33
3.3.13 getAllL2Topologies 3-35
3.3.14 getAllQoSTemplateNames 3-38
3.3.15 getAllQoSTemplates 3-38
3.3.16 getQoSTemplate 3-41
3.3.17 getAllEMSandMEClearedAlarms 3-41
3.3.18 getAllEMSSystemClearedAlarms 3-42
3.3.19 createServerTrailLink 3-43
3.3.20 deleteServerTrailLink 3-46
3.3.21 addNetworkPartition 3-47
3.3.22 addMultiLayerSubnetwork 3-47
3.3.23 deleteNetworkPartition 3-49

3.3.24 deleteMultiLayerSubnetwork 3-49
3.3.25 getAllSupportedMEModels 3-50
3.4 EmsSession::EmsSession_I 3-50
3.4.1 getEventChannel 3-50
3.4.2 getManager 3-51
3.4.3 getSupportedManagers 3-52
3.5 emsSessionFactory::EmsSessionFactory_I 3-52
3.5.1 getEmsSession 3-52
3.5.2 getEmsPublicKey 3-53
3.5.3 getEmsPublicKeyPair 3-54
3.6 equipment::EquipmentInventoryMgr_I 3-55
3.6.1 getAllEquipment 3-55
3.6.2 getAllEquipmentNames 3-63
3.6.3 getAllSupportedPTPs 3-63
3.6.4 getAllSupportedPTPNames 3-71
3.6.5 getAllSupportingEquipment 3-72
3.6.6 getAllSupportingEquipmentNames 3-72
3.6.7 getContainedEquipment 3-73
3.6.8 getEquipment 3-74
3.6.9 provisionEquipment 3-75
3.6.10 setAlarmReportingOff 3-76
3.6.11 setAlarmReportingOn 3-77
3.6.12 unprovisionEquipment 3-78
3.7 maintenanceOps::MaintenanceMgr_I 3-79
3.7.1 getActiveMaintenanceOperations 3-79
3.7.2 performMaintenanceOperation 3-80
3.8 managedElementManager::ManagedElementManager_I 3-82
3.8.1 getAllActiveAlarms 3-82
3.8.2 getAllSNCs 3-83
3.8.3 getAllManagedElements 3-85
3.8.4 getAllManagedElementNames 3-87
3.8.5 getAllPTPs 3-88
3.8.6 getAllPTPNames 3-89
3.8.7 getContainedInUseTPs 3-90
3.8.8 getContainedInUseTPNames 3-92
3.8.9 getContainedPotentialTPs 3-92
3.8.10 getContainedPotentialTPNames 3-94
3.8.11 getContainingSubnetworkNames 3-94
3.8.12 getManagedElement 3-95

3.8.13 getAllCtmMEInternalPatchcord 3-96
3.8.14 setAdditionalInfo 3-96
3.8.15 getTP 3-98
3.8.16 getAssociatedTopologicalLinks 3-99
3.8.17 setTPData 3-100
3.8.18 getAllSrcPTPsForSNCProvisioning 3-102
3.8.19 getAllSrcCTPsForSNCProvisioning 3-103
3.8.20 getAllSrcCTPsForVCATMemberSNCProvisioning 3-104
3.8.21 getAllDestPTPsForSNCProvisioning 3-105
3.8.22 getAllDestCTPsForSNCProvisioning 3-106
3.8.23 getAllDestCTPsForVCATMemberSNCProvisioning 3-107
3.8.24 getAllSrcFTPsForSNCProvisioning 3-108
3.8.25 getAllDestFTPsForSNCProvisioning 3-109
3.8.26 forceNEResync 3-110
3.8.27 getAllTopologicalLinksForME 3-111
3.8.28 getAllRolls 3-111
3.8.29 getAllRollNames 3-113
3.8.30 getRoll 3-114
3.8.31 getRollToPTPs 3-114
3.8.32 getRollToCTPs 3-115
3.8.33 getRollFromCTPs 3-116
3.8.34 setPortName 3-117
3.8.35 getPortName 3-118
3.8.36 getAllClearedAlarms 3-118
3.8.37 getAllCrossConnections 3-120
3.8.38 getAllSoftwareDownloadJobs 3-121
3.8.39 setMEAdminState 3-122
3.9 multiLayerSubnetwork::MultiLayerSubnetworkMgr_I 3-123
3.9.1 getAllEdgePoints 3-123
3.9.2 getAllEdgePointNames 3-124
3.9.3 getAllManagedElement 3-126
3.9.4 getAllManagedElementNames 3-126
3.9.5 getAllSubnetworkConnections 3-127
3.9.6 getAllSubnetworkConnectionsWithTP 3-128
3.9.7 getAllSubnetworkConnectionNames 3-130
3.9.8 getAllSubnetworkConnectionNamesWithTP 3-130
3.9.9 getAllTopologicalLinks 3-131
3.9.10 getAllTopologicalLinkNames 3-132
3.9.11 getMultiLayerSubnetwork 3-133
3.9.12 getCTPsAndTopologicalLinks 3-134

3.9.13 getSNC 3-135
3.9.14 getSNCsByUserLabel 3-136
3.9.15 getTopologicalLink 3-137
3.9.16 createAndActivateSNCFromUserLabel 3-138
3.9.17 CreateAndActivateSNC 3-146
3.9.18 (Deprecated) createAndActivateProtectedOchccFromUserLabel 3-147
3.9.19 addDrops 3-148
3.9.20 deactivateAndDeleteSNC 3-149
3.9.21 getVCATSNC 3-150
3.9.22 getAllVCATSNCs 3-152
3.9.23 getAllVCATSNCNames 3-153
3.9.24 getAllVCATMemberSNCs 3-154
3.9.25 getAllVCATMemberSNCNames 3-155
3.9.26 createVCAT 3-156
3.9.27 deleteVCAT 3-158
3.9.28 addMemberSNCsToVCAT 3-158
3.9.29 deleteMemberSNCsFromVCAT 3-159
3.9.30 deleteAllMemberSNCFromVCAT 3-160
3.9.31 getValidSubnetworkConnectionsForMerge 3-161
3.9.32 upgrade 3-162
3.9.33 createRoll 3-163
3.9.34 deleteRoll 3-166
3.9.35 cancelRoll 3-167
3.9.36 forceValidSignal 3-167
3.9.37 completeRoll 3-168
3.9.38 finishRoll 3-169
3.9.39 getAllRolls 3-169
3.9.40 getAllRollNames 3-170
3.9.41 getAssociatedRollNamesForSNC 3-171
3.9.42 getAssociatedSNCNamesForRoll 3-171
3.9.43 createAndActivateSNCOnDRI 3-172
3.9.44 addManagedElement 3-174
3.9.45 deleteManagedElement 3-179
3.10 L2 Topology and ML VLAN for ML-Series Ethernet Cards 3-180
3.10.1 createMLVLAN 3-180
3.10.2 modifyMLVLANAddFrontPorts 3-183
3.10.3 modifyMLVLANDropFrontPorts 3-184
3.10.4 modifyMLVLANPortState 3-185
3.10.5 modifyMLVLANRSTPState 3-186
3.10.6 modifyMLVLANTPQOS 3-187

3.10.7 modifyMLVLANCustomerIDServiceID 3-188
3.10.8 getMLVLAN 3-189
3.10.9 deleteMLVLAN 3-189
3.10.10 getAllMLVLANs 3-190
3.10.11 getAllMLVLANNames 3-191
3.10.12 getAllAssociatedSNCs 3-192
3.10.13 getL2Topology 3-193
3.10.14 getAssociatedL2TopologyForSNC 3-193
3.10.15 getAssociatedL2TopologyForMLVLAN 3-194
3.10.16 createL2Topology 3-195
3.10.17 deleteL2Topology 3-203
3.10.18 modifyL2Topology 3-204
3.10.19 getAllAvailableMLEquipmentOrTPNames 3-204
3.11 E-Series VLAN Interfaces 3-205
3.11.1 createVLAN 3-206
3.11.2 deleteVLAN 3-206
3.11.3 getAllVLANs 3-207
3.12 nmsSession::NmsSession_I 3-208
3.12.1 historyPMDataCompleted 3-208
3.12.2 historyPMDataFailed 3-208
3.12.3 alarmLossOccurred 3-209
3.13 performance::PerformanceManagementMgr_I 3-210
3.13.1 disablePMData 3-210
3.13.2 enablePMData 3-211
3.13.3 setTCATPParameter 3-212
3.13.4 getTCATPParameter 3-267
3.13.5 getCTMHistoryPMData 3-268
3.13.6 getCTMValidHistoryPMData 3-270
3.13.7 deleteHistoryPMDataFile 3-270
3.13.8 getTPHistoryPMData 3-271
3.13.9 getHoldingTime 3-274
3.13.10 getAllCurrentPMData 3-274
3.13.11 getMEPMcapabilities 3-276
3.14 protection::ProtectionMgr_I 3-276
3.14.1 getAllProtectedTPNames 3-277
3.14.2 getAllProtectionGroups 3-277
3.14.3 getProtectionGroup 3-279
3.14.4 performProtectionCommand 3-280
3.14.5 createProtectionGroup 3-283

3.14.6 deleteProtectionGroup 3-286
3.14.7 getAssociatedProtectionGroup 3-287
3.15 session::Session_I 3-287
3.15.1 endSession 3-288
3.15.2 ping 3-288
3.16 mtnmVersion::Version_I 3-289
3.16.1 getVersion 3-289
3.17 Transmission Descriptor Interfaces 3-289
3.17.1 getAllTransmissionDescriptors 3-290
3.17.2 getAllTransmissionDescriptorsNames 3-291
3.17.3 getTransmissionDescriptor 3-292
3.17.4 getAssociatedTPs 3-292
3.17.5 createTransmissionDescriptor 3-293
3.17.6 deleteTransmissionDescriptor 3-294
3.18 Software and Data Manager Interface 3-295
3.18.1 backupME 3-295
3.18.2 getMEBackupStatus 3-296
3.18.3 abortMEBackup 3-297
3.18.4 getBackupList 3-297
3.18.5 activateSoftwareOnME 3-298
3.18.6 revertSoftwareOnME 3-299
3.18.7 restoreME 3-300
3.18.8 getMERestoreStatus 3-300
3.18.9 abortMERestore 3-301
3.18.10 getAllSupportedMESoftwareVersions 3-302
3.18.11 downloadMESoftware 3-303
3.18.12 addSupportedMESoftwareVersion 3-304
3.18.13 getSWDownloadStatus 3-306

APPENDIX A **Notifications, Fault Management, and Probable Cause Mapping** A-1
A.1 Overview of Notifications A-1
A.2 Resolving the Notification Service A-2
A.3 Tips to Improve Notification Throughput A-2
A.4 Supported Events A-3
A.4.1 Object Creation A-3
A.4.2 Object Deletion A-4
A.4.3 Attribute Value Change A-5
A.4.4 State Change Event A-6
A.4.5 Protection Switch A-8

A.4.6 Threshold Crossing Alert A-9
A.4.7 Alarm A-10
A.4.8 Heartbeat Event A-11
A.4.9 Backup Status Event A-11
A.5 Probable Cause Mapping A-12
A.5.1 TMF-Defined Probable Cause A-12
A.5.2 CTM GateWay/CORBA Mapping—EMS A-13
A.5.3 CTM GateWay/CORBA Mapping—MGX 8880/8850/8830 A-13
A.5.4 CTM GateWay/CORBA Mapping—ONS 15216 A-24
A.5.5 CTM GateWay/CORBA Mapping—ONS 15305 and ONS 15305 CTC A-26
A.5.6 CTM GateWay/CORBA Mapping—ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH A-29
A.5.7 CTM GateWay/CORBA Mapping—ONS 15454-M2 and ONS 15454-M6 A-50
A.5.8 CTM GateWay/CORBA Mapping—ONS 15530 A-50
A.5.9 CTM GateWay/CORBA Mapping—ONS 15540 A-54
A.5.10 Optical Monitoring Alarms—ONS 15530 and ONS 15540 A-57
A.5.11 CTM GateWay/CORBA Mapping—ONS 15600 SONET A-59
A.5.12 CTM GateWay/CORBA Mapping—ONS 15600 SDH A-68

APPENDIX B **Server Administration and Configuration** B-1
B.1 Creating an OSS Client Profile for CTM GateWay/CORBA B-1
B.2 Deleting an OSS Client Profile for CTM GateWay/CORBA B-2
B.3 Viewing Currently Logged In CTM GateWay/CORBA OSS Users B-2
B.4 Logging Out CTM GateWay/CORBA OSS Users B-3
B.5 Using Encryption Between the OSS Client and CTM GateWay/CORBA B-3
B.6 Using Multiple Naming Servers B-4
B.7 Naming Conventions for Published CTM GateWay/CORBA Objects B-5
B.8 Location of the Naming Service IOR File B-6
B.9 Useful Debugging Utilities for Resolving Naming Service-Related Issues B-6
B.10 Configuring CTM GateWay/CORBA B-6
B.11 Using the CLI to Start and Stop CTM GateWay/CORBA B-8
B.12 Configuring Secure Socket Layer for CTM GateWay/CORBA B-9
B.12.1 Generating the Server-Side Certificate B-9
B.12.2 Generating the Client-Side Certificate B-10
B.12.3 Adding the Client-Side Certificate to the Server-Side Keystore B-10
B.12.4 Adding the Server-Side Certificate to the Client-Side Keystore B-11
B.12.5 Configuring the Client-Side Properties B-11
B.13 Installation Program B-12

B.14 CTM R9.1-to-CTM R9.2 Migration B-12

APPENDIX C **OSS Use Cases and Client Development** C-1

C.1 Use Cases to Resynchronize the NMS from the EMS C-1

C.2 Developing a CTM GateWay/CORBA Client C-4

C.2.1 Recommended Tools C-4

C.2.2 Sample Code in Java C-5

C.2.3 Running the Client C-11

C.2.4 Known Interoperability Issues When Using Orbix C-12

APPENDIX D **Equipment List for NEs** D-1

D.1 Equipment List for MDS 9000 D-1

D.2 Equipment List for MGX 8830/8850/8880 D-2

D.3 Equipment List for ONS 15216 D-2

D.4 Equipment List for ONS 15305 D-4

D.5 Equipment List for ONS 15305 CTC D-4

D.6 Equipment List for ONS 15310 CL D-5

D.7 Equipment List for ONS 15310 MA SDH D-5

D.8 Equipment List for ONS 15310 MA SONET D-5

D.9 Equipment List for ONS 15327 D-6

D.10 Equipment List for ONS 15454 SDH D-6

D.11 Equipment List for ONS 15454 SONET D-9

D.12 Equipment List for ONS 15454-M2 D-13

D.13 Equipment List for ONS 15454-M6 D-13

D.14 Equipment List for ONS 15530 D-13

D.15 Equipment List for ONS 15540 ESP D-26

D.16 Equipment List for ONS 15540 ESPx D-32

D.17 Equipment List for ONS 15600 SDH D-34

D.18 Equipment List for ONS 15600 SONET D-35

D.19 Equipment List for Unmanaged NEs D-36

Preface

This section explains the objectives, intended audience, and organization of this guide and describes the conventions that convey instructions and other information.

Objectives

This guide provides a detailed description of the northbound Common Object Request Broker Architecture (CORBA) interfaces supported by Cisco Transport Manager (CTM). This guide serves as a reference for developers of Operations Support System (OSS) applications that interface with CTM GateWay/CORBA Release 9.2.

Use this guide to:

- Learn about the northbound CORBA interfaces supported by CTM R9.2
- Understand the input parameters, output parameters, and attributes associated with CTM GateWay/CORBA

Audience

The primary audience for this guide includes network operations personnel and system administrators. This guide assumes the reader is familiar with the following products and topics:

- Cisco ONS 15000-series products
- Cisco MDS 9000 Multilayer Director and Fabric Switch
- Cisco MGX 8880 Media Gateway
- Voice-enabled Cisco MGX 8850 Multiservice Switch
- Voice-enabled Cisco MGX 8830 Multiservice Switch
- Basic internetworking terminology and concepts
- Network topology and protocols
- Microsoft Windows Vista, Windows 2000 Professional, Windows XP Professional, and Windows Server 2003 Terminal Server
- Sun Microsystems Solaris administration
- Oracle database administration

- Object Management Group (OMG) CORBA, notification service, and naming service
- Telecommunication Management Network (TMN) architecture model

Conventions

This guide uses the following conventions:

Item	Convention
Commands and keywords	**boldface font**
Displayed session and system information	`screen font`
Information that the user must enter	**`boldface screen font`**
Variables that the user must supply	*`italic screen font`*
Menu items and button names	**boldface font**
Selecting a menu item	**Option > Network Preferences**

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.

Tip Means *the information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.

Related Documentation

Note You can access the most current CTM R9.2 documentation online at http://www.cisco.com/en/US/products/sw/opticsw/ps2204/tsd_products_support_series_home.html.

The CTM documentation set comprises the following guides:

- *Release Notes for Cisco Transport Manager Release 9.2*—Describes the caveats for CTM.
- *Cisco Transport Manager Release 9.2 Installation Guide*—Explains how to install CTM and how to upgrade from previous releases.

- *Cisco Transport Manager Release 9.2 User Guide*—Describes how to use the CTM software, which consists of user applications and tools for network discovery, network configuration, connection management, fault management, system administration, and security management.
- *Cisco Transport Manager Release 9.2 GateWay/CORBA User Guide and Programmer Manual*—This guide.
- *Cisco Transport Manager Release 9.2 Database Schema*—Describes the database schema that CTM uses to store information in a Structured Query Language (SQL) database such as the Oracle database. The document is designed for users who need to create their own reports without using CTM.
- *Cisco Transport Manager Release 9.2 High Availability Installation Guide*—Explains how to install CTM in a high availability (HA) environment.

Note To obtain the *Cisco Transport Manager Release 9.2 High Availability Installation Guide*, contact your Cisco account representative.

- *Cisco Transport Manager Release 9.2 ML Provisioning Methodology*—Describes the methodology that CTM uses to provision ML-series cards.
- *Cisco Transport Manager Release 9.2 Basic External Authentication*—Describes how CTM supports basic external authentication.
- *Migration Matrix for CTM Service Pack Releases*—Describes the migration matrix for CTM service pack releases.

The following table lists the related NE hardware documentation.

Table 1 Related Cisco NE Documentation

Cisco Product	Document Title
Cisco MDS 9000 Multilayer Director and Fabric Switch	• *Cisco MDS 9000 Family Command Reference* • *Cisco MDS 9000 Family Configuration Guide* • *Cisco MDS 9xxx Series Hardware Installation Guide* • *Cisco MDS 9000 Family Fabric Manager Configuration Guide* • *Cisco MDS SAN-OS Release Compatibility Matrix for IBM SAN Volume Controller Software for Cisco MDS 9000* • *Cisco MDS 9xxx Family CIM Programming Reference Guide* • *Cisco MDS 9000 Family System Messages Guide* • *Cisco Security Advisory: OpenSSL Vulnerability in the Cisco MDS 9000 Family* • *Cisco MDS 9000 Family Software Upgrade and Downgrade Guide* • *Cisco MDS 9000 Family Fabric Manager User Guide* • *Release Notes for the Cisco MDS 9xxx Family*
Cisco MGX 8880 Voice Gateway Voice-enabled MGX 8850/8830 Multiservice Switch	• *Release Notes for Cisco MGX 8880 Media Gateway* • *Cisco MGX 8800/8900 Hardware Installation Guide, Releases 2 - 5.2* • *Regulatory Compliance and Safety Information for the Cisco Multiservice Switch and Media Gateway Products (MGX, BPX, and SES)* • *Cisco ATM Services (AXSM) Configuration Guide and Command Reference for MGX Switches, Release 5.2* • *Cisco MGX 8800/8900 Series Command Reference, Release 5.2* • *Cisco MGX 8800/8900 Series Configuration Guide, Release 5.2* • *Cisco RPM-PR Documentation* • *Cisco RPM-XF Documentation* • *Cisco Voice Switch Service Module (VXSM) Release 5.3 Documentation* • *Cisco Voice Switch Service Module (VXSM) Release 5.2 Documentation* • *Cisco Voice Interworking Service Module (VISM) Release 3.3 Documentation*

Table 1 Related Cisco NE Documentation (continued)

Cisco Product	Document Title
ONS 15216	• *Cisco ONS 15216 EDFA Operations Guide* • *Cisco ONS 15216 EDFA2 Operations Guide* • *Cisco ONS 15216 EDFA3 Operations Guide* • *Cisco ONS 15216 FlexLayer User Guide* • *Cisco ONS 15216 OSC-1510 User Guide* • *Cisco ONS 15216 System Dispersion Compensation Unit User Guide* • *Installing Cisco ONS 15216 100 GHz DWDM Filters* • *Installing Cisco ONS 15216 OADMs* • *Release Notes for the Cisco ONS 15216 EDFA* • *Release Notes for the Cisco ONS 15216 OADM* • *Upgrade Procedures for the Cisco ONS 15216 EDFA2*
ONS 15305 (non-CTC-based)	• *Cisco ONS 15305 Installation and Operations Guide* • *Cisco ONS 15305 Quick Installation Guide* • *Release Notes for the Cisco ONS 15305*
ONS 15305 (CTC-based)	• *Cisco ONS 15305 Cisco Transport Controller Operations Guide, R5.0* • *Release Notes for Cisco ONS 15305 Release 3.0*
ONS 15310 CL and ONS 15310 MA	• *Cisco ONS 15310 Network Defaults* • *Cisco ONS 15310-CL and Cisco ONS 15310-MA Ethernet Card Software Feature and Configuration Guide* • *Cisco ONS 15310-CL and Cisco ONS 15310-MA Procedure Guide* • *Cisco ONS 15310-CL and Cisco ONS 15310-MA Reference Manual* • *Cisco ONS 15310-CL and Cisco ONS 15310-MA Troubleshooting Guide* • *Release Notes for the Cisco ONS 15310*
ONS 15327	• *Cisco ONS 15327 Procedure Guide* • *Cisco ONS 15327 Reference Manual* • *Cisco ONS 15327 Software Upgrade Guide* • *Cisco ONS 15327 Troubleshooting Guide* • *Cisco ONS 15454 and Cisco ONS 15327 TL1 Command Guide* • *Cisco ONS 15454 and Cisco ONS 15327 TL1 Command Quick Reference Guide* • *Release Notes for the Cisco ONS 15327*

Table 1 Related Cisco NE Documentation (continued)

Cisco Product	Document Title
ONS 15454 SDH	• *Cisco ONS 15454 SDH Procedure Guide* • *Cisco ONS 15454 DWDM Procedure Guide* • *Cisco ONS 15454 SDH Reference Manual* • *Cisco ONS 15454 SDH Software Upgrade Guide* • *Cisco ONS 15454 SDH Troubleshooting Guide* • *Cisco ONS 15454 SONET/SDH ML-Series Multilayer Ethernet Card Software Feature and Configuration Guide* • *Release Notes for the Cisco ONS 15454 SDH*
ONS 15454 SONET	• *Cisco ONS 15454 and Cisco ONS 15327 TL1 Command Guide* • *Cisco ONS 15454 and Cisco ONS 15327 TL1 Command Quick Reference Guide* • *Cisco ONS 15454 Procedure Guide* • *Cisco ONS 15454 Reference Manual* • *Cisco ONS 15454 Software Upgrade Guide* • *Cisco ONS 15454 SONET/SDH ML-Series Multilayer Ethernet Card Software Feature and Configuration Guide* • *Cisco ONS 15454 Troubleshooting Guide* • *Release Notes for the Cisco ONS 15454*
ONS 15530	• *Cisco ONS 15530 Configuration Guide and Command Reference* • *Cisco ONS 15530 Hardware Installation Guide* • *Cisco ONS 15530 MIB Quick Reference* • *Cisco ONS 15530 Optical Transport Turn-up and Test Guide* • *Cisco ONS 15530 Planning and Design Guide* • *Cisco ONS 15530 System Alarms and Error Messages* • *Network Management for the Cisco ONS 15530* • *Quick Reference for the Cisco ONS 15530 TL1 Commands* • *Release Notes for the Cisco ONS 15530*
ONS 15540 ESP	• *Cisco ONS 15540 ESP Configuration Guide and Command Reference* • *Cisco ONS 15540 ESP Hardware Installation Guide* • *Cisco ONS 15540 ESP MIB Quick Reference* • *Cisco ONS 15540 ESP Planning and Design Guide* • *Cisco ONS 15540 ESP Troubleshooting Guide* • *Network Management for the Cisco ONS 15540 ESP* • *Optical Transport Turn-Up and Test Guide* • *Regulatory Compliance and Safety Information for the Cisco ONS 15500 Series* • *Release Notes for the Cisco ONS 15540 ESP* • *ROMMON and Functional Image Release Notes*

Table 1 Related Cisco NE Documentation (continued)

Cisco Product	Document Title
ONS 15540 ESPx	• *Cisco ONS 15540 ESPx Cleaning Procedures for Fiber Optic Connections* • *Cisco ONS 15540 ESPx Configuration Guide and Command Reference* • *Cisco ONS 15540 ESPx Hardware Installation Guide* • *Cisco ONS 15540 ESPx MIB Quick Reference* • *Cisco ONS 15540 ESPx Planning and Design Guide* • *Cisco ONS 15540 ESPx System Alarms and Error Messages* • *Network Management for the Cisco ONS 15540 ESPx* • *Optical Transport Turn-Up and Test Guide* • *Quick Reference for the Cisco ONS 15540 ESP and Cisco ONS 15540 ESPx TL1 Commands* • *Regulatory Compliance and Safety Information for the Cisco ONS 15500 Series* • *Release Notes for the Cisco ONS 15540 ESPx* • *ROMMON and Functional Image Release Notes*
ONS 15600 SDH	• *Cisco ONS 15600 SDH Procedure Guide* • *Cisco ONS 15600 SDH Reference Manual* • *Cisco ONS 15600 SDH TL1 Test Access* • *Cisco ONS 15600 SDH Troubleshooting Guide* • *Release Notes for the Cisco ONS 15600 SDH*
ONS 15600 SONET	• *Cisco ONS 15600 Procedure Guide* • *Cisco ONS 15600 Reference Manual* • *Cisco ONS 15600 TL1 Command Guide* • *Cisco ONS 15600 Troubleshooting Guide* • *Release Notes for the Cisco ONS 15600*

The following related documentation may also be used as reference:

- Telcordia GR-253-CORE: Synchronous Optical Network (SONET) Transport Systems: Common Generic Criteria
- Telcordia GR-815-CORE: Generic Requirements for Network Elements/Network Systems (NE/NS) Security
- Telcordia GR-820-CORE: Generic Digital Transmission Surveillance
- Telcordia GR-831-CORE: Operations Applications Messages
- Telcordia GR-2998-CORE: Generic Requirements for Wavelength Division Multiplexing (WDM) Element Management Systems
- Telcordia GR-3000-CORE: Generic Requirements for SONET Element Management Systems
- ITU-T G.774: Synchronous Digital Hierarchy (SDH)—Management information for the network element view
- ITU-T G.784: Synchronous Digital Hierarchy (SDH) management

- ITU-T G.826: Error performance parameters and objectives for international, constant bit rate digital paths at or above the primary rate
- ITU-T G.829: Error performance events for SDH multiplex and regenerator sections
- ITU-T G.831: Management Capabilities of Transport Networks based on the synchronous digital hierarchy (SDH)
- ITU-T G.872: Architecture of optical transport networks
- ITU-T M.3010: Principles for a Telecommunications management network
- ITU-T M.3100: Generic network information model
- ITU-T M.3400: TMN Management Functions
- ITU-T X.721: Information technology–Open Systems Interconnection–Structure of management information: Definition of management information
- ITU-T X.731: Information technology–Open Systems Interconnection–Systems Management: State management function
- ITU-T X.733: Information technology–Open Systems Interconnection–Systems Management: Alarm reporting function
- ITU-T X.734: Information technology–Open Systems Interconnection–Systems Management: Event report management function
- ITU-T X.735: Information technology–Open Systems Interconnection–Systems Management: Log control function
- ITU-T X.736: Information technology–Open Systems Interconnection–Systems Management: Security alarm reporting function
- TMF 513: TM Forum Business Agreement, version 1.02
- TMF 513: Multi Technology Network Management Business Agreement, version 2.0
- TMF 608: Multi Technology Network Management Information Agreement, version 0.4
- TMF 608: Multi Technology Network Management Information Agreement, version 2.0
- TMF 814: Multi Technology Network Management Solution Set, version 0.2
- TMF 814: Multi Technology Network Management Solution Set, version 2.0
- TMF 814A: Multi Technology Network Management Solution Set, TM Forum Implementation (IS) Template Guidelines, version 2.0

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

CHAPTER 1

Overview of CTM GateWay/CORBA

This chapter provides an overview of CTM GateWay/CORBA. It includes the following information:

- 1.1 Features of CTM GateWay/CORBA, page 1-3
- 1.2 CTM GateWay/CORBA Fault Management, page 1-3
- 1.3 Overview of CORBA Services, page 1-4
- 1.4 Information Model, page 1-7
- 1.5 IDL-Related Changes, page 1-8
- 1.6 Functionalities Not Supported in R9.2, page 1-9
- 1.7 Summary, page 1-9

CTM is a highly advanced optical transport domain manager. Open interfaces to Operations Support Systems (OSSs) using industry-standard protocols make CTM a flexible building block in legacy as well as next-generation OSS infrastructures. The CTM GateWay/CORBA product is a CORBA northbound interface that complies with the TMF CORBA specification for network management layer-element management layer (NML-EML) interfaces.

A programming technology for distributed computing, CORBA enables components of various application programs to communicate with one another regardless of their implementation characteristics—programming language, operating system, machine, location, and so on. CTM GateWay/CORBA allows service providers to implement a programmatic interface between CTM and various OSS tools in the back office. Such interfaces enable next-generation flow-through equipment and circuit management, including inventory, provisioning, and assurance functions.

For each NE that CTM manages, the CTM GateWay/CORBA option provides higher-layer management systems with fault, inventory, and performance information, plus configuration and circuit provisioning capabilities. CTM GateWay/CORBA supports up to eight simultaneous sessions with higher-layer management systems.

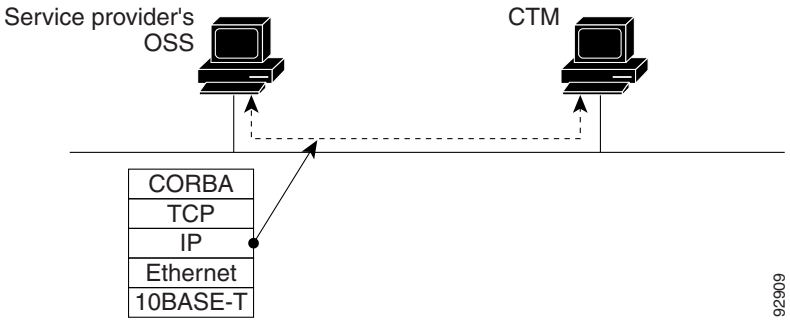
CTM GateWay/CORBA supports the following NEs in CTM R9.2:

- MGX 8880/8850/8830
- ONS 15200
- ONS 15216
- ONS 15305
- ONS 15310 CL
- ONS 15310 MA SDH
- ONS 15310 MA SONET

- ONS 15327
- ONS 15454 SDH
- ONS 15454 SONET
- ONS 15530
- ONS 15540
- ONS 15600 SDH
- ONS 15600 SONET

The following figure illustrates the CTM GateWay/CORBA communications architecture within a service provider's OSS environment.

Figure 1-1 CTM GateWay/CORBA Communications Architecture



The telemanagement community has adopted CORBA as the preferred distribution technology with which to integrate the various components of an OSS. The Multi Technology Network Management Working Group of the TMF has been working to create a specification for an NML-EML interface. CTM GateWay/CORBA is based on the following TMF standards:

- TMF.513 v2.0 (August 2001): Multi Technology Network Management Business Agreement
- TMF.608 v2.0 (August 2001): Multi Technology Network Management Information Agreement
- TMF.814 v3.0 (August 2001): Multi Technology Network Management Solution Set

The TMF.608 document is a protocol-independent information model for the network resources being managed. The TMF.814 document is the CORBA Interface Data Language (IDL) component of the specification; it consists of a set of definitions for specific objects and methods that are available for use at the NML-EML interface for passing information or accomplishing specific actions. However, TMF.814 is a specific interface implementation based on a TMF high-level business agreement described by the TMF.513 document. To complete the document set, TMF.814A recognizes that different parties may provide varying degrees of compliance in their implementations. The TMF.814A document is called the Implementation Agreement, and is simply a form to be filled out by those parties implementing TMF.814 to further describe their specific implementations to those who might use them.

1.1 Features of CTM GateWay/CORBA

CTM GateWay/CORBA supports the following TM.814 modules:

- Inventory
  - Element management system (EMS)
  - Multilayer subnetwork
  - Managed element
  - Physical termination point (PTP)
  - Connection termination point (CTP)
  - Floating termination point (FTP)
  - Subnetwork connection (SNC)
  - Protection group
  - Equipment
  - Equipment holder
  - Topological link
- Alarm notification
- Retrieval of active alarms on objects where inventory is available
- SNC provisioning
- Equipment provisioning
- Protection operations
- Performance operations
- Maintenance operations

Note Performance and inventory application programming interfaces (APIs) do not retrieve any information for out-of-service NEs.

The gateway authenticates the northbound OSS. Once authenticated, the OSS can perform operations and receive alarm notifications.

CTM GateWay/CORBA uses jacORB 2.1.3.7. The corresponding standard is Object Management Group (OMG) CORBA 2.3.

1.2 CTM GateWay/CORBA Fault Management

CTM GateWay/CORBA forwards all the fault notifications received from managed NEs and all CTM alarms to the network management system (NMS) through the notification service. Information about the notification service is available on the OMG website at www.omg.org.

Note This website is copyright © 1997–2009, Object Management Group, Inc.

The NMS retrieves all active alarms on the EMS and on managed element objects. The NMS filters alarms based on probable cause, perceived severity, and managed element name.

1.3 Overview of CORBA Services

CTM GateWay/CORBA uses two CORBA services—naming service and notification service. The naming service is required for CTM GateWay/CORBA to function. If the notification service is not available, alarm and event notifications are not forwarded to the OSS. By default, the naming and notification services are installed on the CTM server during the GateWay/CORBA installation. However, the CORBA services can be located anywhere as long as they are accessible from the CTM server on which CTM GateWay/CORBA is installed. The OSS application(s) must also use the same instances of the naming service and the notification service.

1.3.1 Naming Service

CTM GateWay/CORBA and the other CTM server processes must use a naming service in order to operate. CTM GateWay/CORBA includes the OpenFusion naming service product. By default, CTM GateWay/CORBA is configured to use OpenFusion. However, you can configure CTM GateWay/CORBA and the CTM server to use a different centralized naming service. For more information, see B.6 Using Multiple Naming Servers, page B-4.

1.3.2 Notification Service

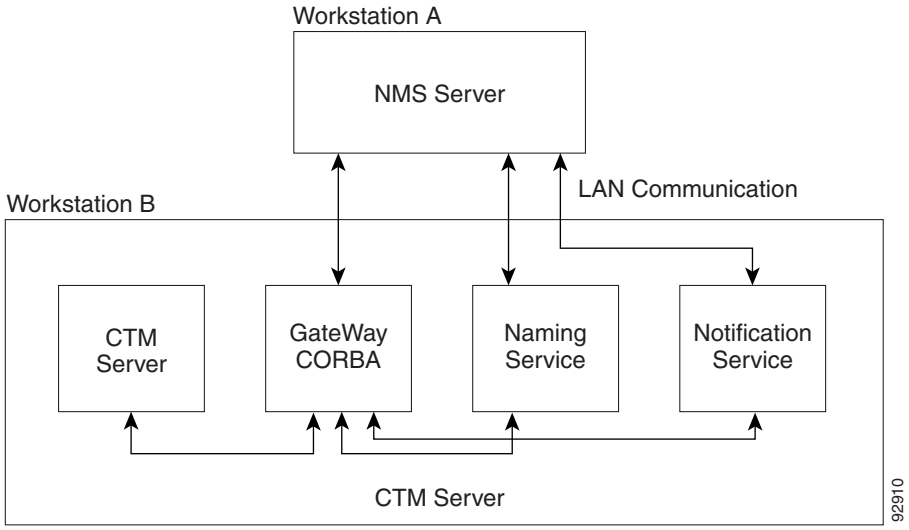
CTM GateWay/CORBA must use a CORBA notification service to report faults and events. CTM GateWay/CORBA includes the PrismTech OpenFusion notification service product, and is configured by default to use it. As with the naming service, you can configure CTM GateWay/CORBA to use a different notification service. For more information, see A.1 Overview of Notifications, page A-1.

The following scenarios describe possible configurations of the naming and notification services.

Scenario 1: Naming and Notification Services Installed on the CTM Server

The naming and notification services are installed on the CTM server workstation as shown in the following figure.

Figure 1-2 Naming and Notification Services on the CTM Server Workstation

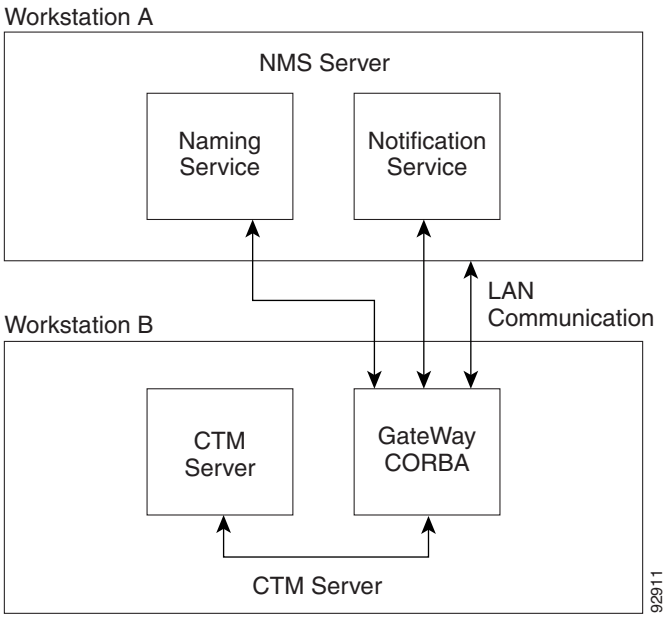


The notification service must be installed on the CTM server workstation with the CTM server processes. CTM GateWay/CORBA is configured to use this notification service. The NMS processes must be configured to use the notification and naming services on the CTM server workstation.

Scenario 2: Naming and Notification Services Installed on the NMS Workstation

The naming and notification services are installed on the NMS workstation as shown in the following figure.

Figure 1-3 Naming and Notification Services on the NMS Workstation

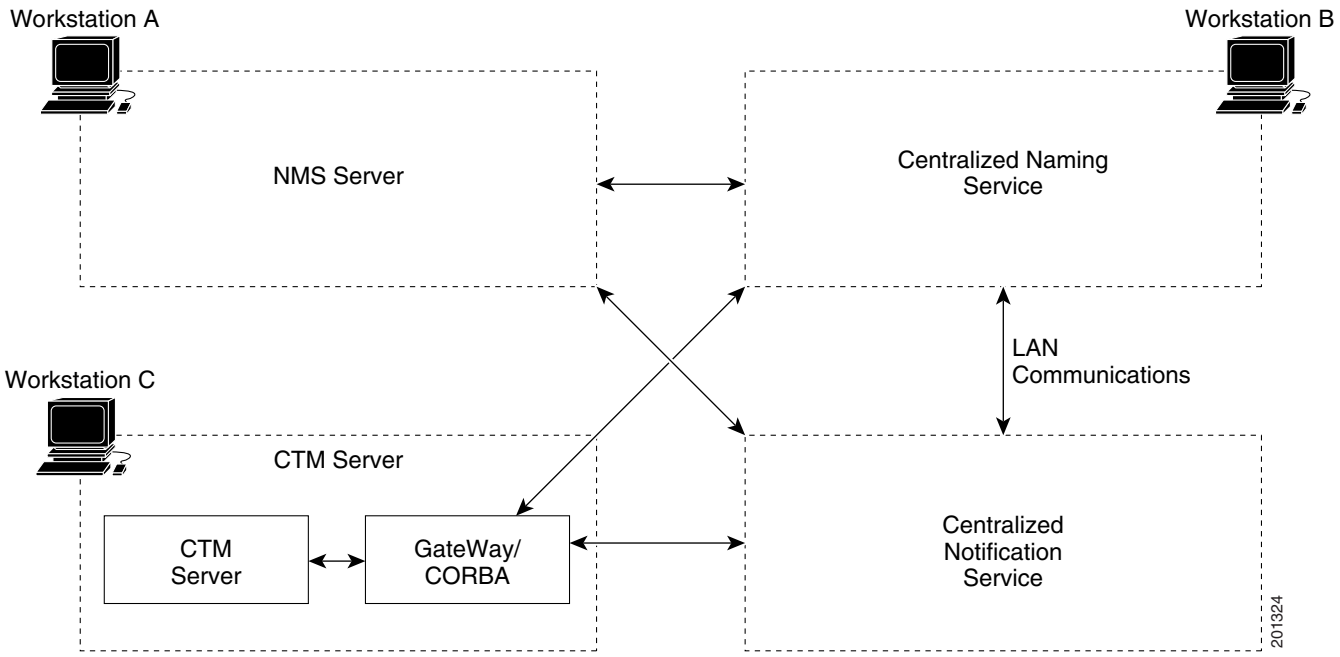


The CTM server processes and CTM GateWay/CORBA must be configured to use the naming and notification services on the NMS workstation.

Scenario 3: Naming and Notification Services Installed on Independent Servers

The naming and notification services are installed on independent servers as shown in the following figure.

Figure 1-4 Naming and Notification Services on Independent Servers



The CTM server processes, CTM GateWay/CORBA, and the NMS processes must be configured to use the naming and notification services on independent workstations.

1.4 Information Model

The information model used by the CTM GateWay/CORBA northbound interface is based on the TMF NML-EML CORBA interface—TMF.814 v3.0. This standard is available to TMF members from the TMF website. The information model allows the EMS to provide an abstraction layer between the EMS and the NMS. This standard is generic and is, therefore, applicable to many different EMS and NE implementations. Customers implementing a TMF-compliant CORBA interface between a specific NMS and CTM must use the CTM GateWay/CORBA product, and will require specific information on what API features are implemented and how those features should be used.

1.5 IDL-Related Changes

The following table lists the IDL-related changes that occurred between two given releases.

Table 1-1 Summary of IDL-Related Changes

Release Details	Change
From R9.1 to R9.2	No change
From R9.0 to R9.1	• managedElementManager.idl extended with new API void getAllCtmMEInternalPatchord(in globaldefs::NamingAttributes_T managedElementName, out globaldefs::NVSList_T patches) raises (globaldefs::ProcessingFailureException); • performance.idl extended with new API void getCTMValidHistoryPMData(in nmsSession::NmsSession_I client, in PMTPSelectList_T pmTPSelectList, in PMParameterNameList_T pmParameters, in globaldefs::Time_T startTime, in globaldefs::Time_T endTime) raises(globaldefs::ProcessingFailureException);

From R8.5 to R9.0	• New enumerated PC_LOCKON, PGT_SPLITTER, PGT_L2_1_PLUS_1 added to protection.idl
	• New enumerated OT_NETWORK_PARTITION added to notifications.idl.
	• New interface type added to multiLayerSubnetwork.idl:
	enum L2InterfaceType_T
	{
	L2IT_UNKNOWN,
	L2IT_FASTETHER,
	L2IT_GIGAETHER,
	L2IT_PORTCHANNEL
	};
	L2InterfaceType_T interfaceType;

1.6 Functionalities Not Supported in R9.2

The following table lists the functionalities which are not supported in release 9.2. The cards or equipment in which the functionalities are not supported are also listed.

Table 1-2 Functionalities Not Supported

Functionalities	Card/Equipment
Historical PM retrieval	• 80-WXC-C • MSNE 30 Shelf • XP_10GE (gz) for LACP, REP
Realtime PM retrieval	• 80-WXC-C • MSNE 30 Shelf • ADM_10G for RMON, IPPMHDLC/LEX • XP_10GE (gz) for E-OAM, LACP, REP • TNC • 40G_MXP_C
Get and set thresholds	80-WXC-C
PPC links	80-WXC-C
Alarm management	XP_10GE (gz) for CFM, REP, E-OAM, LACP

1.7 Summary

As a state-of-the-art EMS, CTM can provide a CORBA-based interface between itself and an NMS. This functionality is provided by a separately purchasable option—CTM GateWay/CORBA.
CTM GateWay/CORBA provides a TMF-compliant CORBA interface in accordance with the TMF NML-EML specification composed of TMF.513, TMF.608, and TMF.814. CTM GateWay/CORBA includes naming and notification services. Customers can use the bundled naming and notification services or their own services in various configurations. CTM GateWay/CORBA brings the service provider one step closer to full OSS automation.

CHAPTER 2

NE- and CTM-Specific Details

This chapter provides specific details about each of the NEs that CTM supports. It also provides details specific to CTM.

The following sections include NE- and CTM-specific details:

- 2.1 NE-Specific Details, page 2-1
- 2.2 CTM-Specific Details, page 2-23

Note In this document, the terms *network element* (NE) and *managed element* (ME) are interchangeable. Also, *card* and *equipment* strings are used interchangeably to refer to equipment.

2.1 NE-Specific Details

2.1.1 ONS 15216

Some NEs in the ONS 15216 family do not provide a management interface. These NEs are considered *passive* because the user enters all data from the CTM client. After the data has been entered, CTM reports these NEs. Other ONS 15216 NEs provide a management interface; these NEs are considered *active*.

See the *Release Notes for Cisco Transport Manager Release 9.2* for the passive and active ONS 15216 software releases that CTM R9.2 supports.

Note All ONS 15216 NEs contain a single shelf and have no removable equipment, except for the ONS 15216 DCU, which contains a chassis and a removable module. The ONS 15216 FlexLayer also has a removable module.

2.1.1.1 PTPs

CTM reports PTPs for ONS 15216 NEs only if a topological link originates from or terminates on the ONS 15216 NE.

2.1.1.2 Topological Links

Topological links are unidirectional for ONS 15216 NEs. Inter-NE links are available between all NEs in the ONS 15216 family. CTM supports links between an ONS 15216 and the following NEs at compatible layer rates:

- ONS 15216
- ONS 15310 CL and ONS 15310 MA SONET
- ONS 15327 (physical, SONET/SDH, and OCH)
- ONS 15454 (physical, SONET/SDH, and OCH)
- ONS 15600 (physical, SONET/SDH, and OCH)

2.1.2 ONS 15305 and ONS 15305 CTC

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15305 software releases that CTM R9.2 supports.

Note NEs with R3.x are called ONS 15305 CTC.

2.1.2.1 Slots

The ONS 15305 and ONS 15305 CTC are single-shelf NEs with four slots.

2.1.2.2 PTPs

The ONS 15305 and ONS 15305 CTC support PTPs associated to WAN, SDH, and PDH ports.

2.1.3 ONS 15310 CL

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15310 CL software releases that CTM R9.2 supports.

2.1.3.1 Slots

The ONS 15310 CL is a single-shelf NE with two slots. Slot 1 is the expansion slot and contains the provisionable data card. Slot 2 is a nonreplicable unit and contains the Control Timing and Cross-Connect (CTX)-CL controller.

2.1.3.2 Equipment

All cards for the ONS 15310 CL R5.0 and later display administration and service states. For earlier software releases, these states are not applicable and CTM displays them as *N/A*. The following tables show the attribute values for these two states.

2.1.3.2.1 Equipment Administration State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. Table 2-1 lists the attribute values for the administration state. Table 2-2 and Table 2-3 list the attribute values for the service state.

Table 2-1 Equipment Administration State (ONS 15310 CL)

Attribute Value	Meaning
In Service	Places the entity In Service.
Auto in Service	Places the entity in Auto in Service. The circuit is out of service until it receives a valid signal for the duration of the soak period, at which time the circuit state changes to In Service. During the soak period, alarms and loopbacks are suppressed but traffic is carried.
Out of Service–Maintenance	Removes the entity from service for maintenance. Alarms are no longer generated.
Out of Service	Removes the entity from service and disables it. Alarms are no longer generated and traffic is not carried.

2.1.3.2.2 Equipment Service State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. The attribute value for the service state is the same as that shown in CTM.

Table 2-2 Equipment Service State–Primary (ONS 15310 CL)

Primary States (PST-PSTQ)	
Attribute Value	**Meaning**
In Service–Normal (IS_NR)	The entity is fully operational and performs as provisioned.
Out of Service–Management (OOS_MA)	The entity has been administratively removed from service.
Out of Service–Autonomous (OOS_AU)	The entity is not operational due to an autonomous event.
Out of Service–Autonomous and Management (OOS_AUMA)	The entity is not operational due to an autonomous event and has been administratively removed from service.

Table 2-3 Equipment Service State–Secondary (ONS 15310 CL)

Secondary States (SST)	
Attribute Value	**Meaning**
Unassigned (UAS)	The entity is not provisioned in the database.
Unequipped (UEQ)	The entity has been removed, leaving an empty slot. It may or may not be assigned (provisioned).
Mismatch of Equipment and Attributes (MEA)	Incorrect equipment is installed. For example, the card plugged in is not compatible with the provisioned card or slot.
Automatic In-Service (AINS)	The entity is in a delay transition (to IS) state. The transition to IS is pending the correction of off-normal conditions on the entity.
Maintenance (MT)	The entity has been manually removed from service for maintenance. It is still capable of performing its provisioned functions. Traffic is still carried.

Table 2-3 Equipment Service State–Secondary (ONS 15310 CL) (continued)

Secondary States (SST)	
Attribute Value	**Meaning**
Disabled (DSBLD)	The entity has been manually removed from service and cannot perform its provisioned functions. All of its provisioned services are disrupted. Traffic is not carried.
Loopback (LPBK)	The loopback command is in effect (any loopback type).

2.1.3.3 Topological Links

All topological links are unidirectional or bidirectional for ONS 15310 CL NEs.

2.1.3.4 CTX Cards

The CTX card is an electrical card that consists of 21 digital signal level 1 (DS1) ports and three DS3/EC1 ports. Two optical interfaces are supported using SFP technology. OC-3/STM-1 and OC-12/STM-4 ports are available.

2.1.3.5 PTPs

All PTPs are bidirectional for the ONS 15310 CL. The ports are always in channelized mode. PTPs support In Service and Out of Service–Maintenance values for the service state attribute.

2.1.3.6 CTPs

All CTPs are in channelized mode for the ONS 15310 CL.

2.1.3.7 SNCs

CTX cards perform synchronous transport signal (STS) and VT1.5 switching. The subnetwork connection (SNC) name cannot exceed 48 characters. The NE enforces the SNC name character limit.

2.1.3.8 Autodiscovery

The ONS 15310 CL supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all information, including the actual name, and an attribute value change (AVC) event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.4 ONS 15310 MA SDH

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15310 MA SDH software releases that CTM R9.2 supports.

2.1.4.1 Slots

The ONS 15310 MA SDH consists of two CTX slots and four Expansion (EXP) I/O slots.

2.1.4.2 Topological Links

All topological links are unidirectional or bidirectional for ONS 15310 MASDH NEs.

2.1.4.3 Cards

The ONS 15310 MA SDH supports the following cards:

- CTX2500
  - E1_21_E3_DS3_3_LINE_CARD—The electrical ports consist of 21 DS1 and 3 DS3 or EC1 ports.
  - E1_63_E3_DS3_3_LINE_CARD—The electrical ports consist of 63 DS1 and 3 DS3 or EC1 ports.
- ML-100T-8
- CE-100T-8

2.1.4.4 PTPs

All PTPs are bidirectional for the ONS 15310 MA SDH. The ports are always in channelized mode. PTPs support In Service and Out of Service–Maintenance values for the service state attribute.

SNC provisioning on E1_21_DS3_3 and E1_63_E3_3 cards uses the following DS3/E3 ports:

- E1_21_DS3_3—Port numbers 22, 23, and 24.
- E1_63_E3_3—Port numbers 64, 65, and 66.

2.1.4.5 Auto Discovery

The ONS 15310 MA SDH supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all information, including the actual name, and an AVC event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.5 ONS 15310 MA SONET

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15310 MA SONET software releases that CTM R9.2 supports.

2.1.5.1 Slots

The ONS 15310 MA SONET consists of two CTX slots and four EXP I/O slots.

2.1.5.2 Topological Links

All topological links are unidirectional or bidirectional for ONS 15310 MA SONET NEs.

2.1.5.3 CTX Cards

The CTX card is an electrical card that consists of 21 DS1 ports and three DS3/EC-1 ports. Two optical interfaces are supported using SFP technology. OC-3/STM-1 and OC-12/STM-4 ports are available.

- DS1_28_DS3_EC1_3_LINE_CARD—The electrical ports consist of 28 DS1 and 3 DS3 or EC1 ports.
- DS1_84_DS3_EC1_3_LINE_CARD—The electrical ports consist of 84 DS1 and 3 DS3 or EC1 ports.

2.1.5.4 PTPs

All PTPs are bidirectional for the ONS 15310 MA SONET. The ports are always in channelized mode. PTPs support In Service and Out of Service–Maintenance values for the service state attribute.

SNC provisioning on DS1_28_DS3_3 and DS1_84_DS3_3 cards uses the following DS3/EC1 ports:

- DS1_28_DS3_3—Port numbers 29, 30, and 31.
- DS1_84_DS3_3—Port numbers 85, 86, and 87.

2.1.5.5 CTPs

All CTPs are in channelized mode for the ONS 15310 MA SONET.

2.1.5.6 SNCs

CTX cards perform STS and VT1.5 switching. The SNC name cannot exceed 48 characters. The NE enforces the SNC name character limit.

2.1.5.7 Auto Discovery

The ONS 15310 MA SONET supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all

information, including the actual name, and an AVC event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.6 ONS 15327

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15327 software releases that CTM R9.2 supports.

2.1.6.1 Slots

The ONS 15327 is a single-shelf NE with eight slots. The Mechanical Interface Card (MIC) A card must occupy slot 8; the MIC B card must occupy slot 7. If an integrated Cross-Connect, Timing, and Control (XTC) card is required, it must occupy slot 5 or slot 6.

2.1.6.2 Equipment

All of the cards in the ONS 15327 R5.0 and later display administration and service states. For earlier software releases, these states are not applicable and CTM displays them as *N/A*.

2.1.6.2.1 Equipment Administration State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. Table 2-4 lists the attribute values for the administration state. Table 2-5 and Table 2-6 list the attribute values for the service state.

Table 2-4 Equipment Administration State (ONS 15327)

Attribute Value	Meaning
In Service	Places the entity In Service.
Auto in Service	Places the entity in Auto in Service. The circuit is out of service until it receives a valid signal for the duration of the soak period, at which time the circuit state changes to In Service. During the soak period, alarms and loopbacks are suppressed but traffic is carried.
Out of Service–Maintenance	Removes the entity from service for maintenance. Alarms are no longer generated.
Out of Service	Removes the entity from service and disables it. Alarms are no longer generated and traffic is not carried.

2.1.6.2.2 Equipment Service State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. The attribute value for the service state is the same as that shown in CTM.

Table 2-5 Equipment Service State–Primary (ONS 15327)

Primary States (PST-PSTQ)	
Attribute Value	**Meaning**
In Service–Normal (IS_NR)	The entity is fully operational and performs as provisioned.
Out of Service–Management (OOS_MA)	The entity has been administratively removed from service.
Out of Service–Autonomous (OOS_AU)	The entity is not operational due to an autonomous event.
Out of Service–Autonomous and Management (OOS_AUMA)	The entity is not operational due to an autonomous event and has been administratively removed from service.

Table 2-6 Equipment Service State–Secondary (ONS 15327)

Secondary States (SST)	
Attribute Value	**Meaning**
Unassigned (UAS)	The entity is not provisioned in the database.
Unequipped (UEQ)	The entity has been removed, leaving an empty slot. It may or may not be assigned (provisioned).
Mismatch of Equipment and Attributes (MEA)	Incorrect equipment is installed. For example, the card plugged in is not compatible with the provisioned card or slot.
Automatic In-Service (AINS)	The entity is in a delay transition (to IS) state. The transition to IS is pending the correction of off-normal conditions on the entity.
Maintenance (MT)	The entity has been manually removed from service for maintenance. It is still capable of performing its provisioned functions. Traffic is still carried.
Disabled (DSBLD)	The entity has been manually removed from service and cannot perform its provisioned functions. All of its provisioned services are disrupted. Traffic is not carried.
Loopback (LPBK)	The loopback command is in effect (any loopback type).

2.1.6.3 Topological Links

All topological links are unidirectional or bidirectional for ONS 15327 NEs.

2.1.6.4 XTC Cards

The XTC-14 card has 14 DS1 ports numbered 1 through 14. Only VT1.5 SNCs can be created on these ports. The XTC-28-3 card has 28 DS1 ports numbered 1 through 28 and 3 DS3 ports numbered 29 through 31. VT1.5 SNCs can be created on DS1 ports, and STS-1 SNCs can be created on DS3 ports.

2.1.6.5 PTPs

All PTPs are bidirectional for the ONS 15327. The ports are always in channelized mode. PTPs support In Service, Out of Service, Out of Service–Maintenance, and Auto in Service values for the service state attribute.

2.1.6.6 CTPs

All CTPs are in channelized mode for the ONS 15327. Alarm monitoring cannot be turned on or off for CTPs.

2.1.6.7 SNCs

XTC cards perform STS and VT1.5 switching. The XTC cards support the total rearrangement of 192 bidirectional STSs from the four high-speed slots (1 to 4), plus 12 bidirectional STSs for XTC module low-speed electrical interfaces. The XTC VT1.5 matrix supports the grooming of 336 bidirectional VT1.5 circuits.

The SNC name cannot exceed 48 characters.

2.1.6.8 Autodiscovery

ONS 15327 autodiscovery is identical to ONS 15454 SONET autodiscovery, described in 2.1.8.10 Autodiscovery, page 2-14.

2.1.7 ONS 15454 SDH

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15454 SDH software releases that CTM R9.2 supports.

2.1.7.1 Shelves

All supported ONS 15454 SDH NEs have exactly one shelf each, with the exception of the ONS 15454 SDH multiservice transport platform (MSTP) R7.0, which can have up to eight shelves per NE.

2.1.7.2 Slots

The ONS 15454 SDH has one shelf view supplemented with the top subrack Electrical Facility Connection Assembly (EFCA) shelf and bottom subrack chassis. The main shelf has 17 slots numbered 1 to 17 and contains cards that carry traffic. The EFCA shelf has 12 additional slots numbered 18 to 29 and contains Front Mount Electrical Connection (FMEC) modules, the ALM/PWR/MIC (A/P/M) module, and the CRFT/TMG/MIC (C/T/M) module. The Timing Communications and Control-International (TCC-I) module must be in slot 7 or slot 11, the A/P/M module must be in slot 23, and the C/T/M module must be in slot 24.

2.1.7.3 Equipment

All cards in the ONS 15454 SDH R5.0 and later display administration and service states. For earlier software releases, these states are not applicable and CTM displays them as *N/A*.

2.1.7.3.1 Equipment Administration State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. Table 2-7 lists the attribute values for the administration state. Table 2-8 and Table 2-9 list the attribute values for the service state.

Table 2-7 Equipment Administration State (ONS 15454 SDH)

Attribute Value	Meaning
In Service	Places the entity In Service.
Auto in Service	Places the entity in Auto in Service. The circuit is out of service until it receives a valid signal for the duration of the soak period, at which time the circuit state changes to In Service. During the soak period, alarms and loopbacks are suppressed but traffic is carried.
Out of Service–Maintenance	Removes the entity from service for maintenance. Alarms are no longer generated.
Out of Service	Removes the entity from service and disables it. Alarms are no longer generated and traffic is not carried.

2.1.7.3.2 Equipment Service State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. The attribute value for the service state is the same as that shown in CTM.

Table 2-8 Equipment Service State–Primary (ONS 15454 SDH)

Primary States (PST-PSTQ)	
Attribute Value	**Meaning**
In Service–Normal (IS_NR)	The entity is fully operational and performs as provisioned.
Out of Service–Management (OOS_MA)	The entity has been administratively removed from service.
Out of Service–Autonomous (OOS_AU)	The entity is not operational due to an autonomous event.
Out of Service–Autonomous and Management (OOS_AUMA)	The entity is not operational due to an autonomous event and has been administratively removed from service.

Table 2-9 Equipment Service State–Secondary (ONS 15454 SDH)

Secondary States (SST)	
Attribute Value	**Meaning**
Unassigned (UAS)	The entity is not provisioned in the database.
Unequipped (UEQ)	The entity has been removed, leaving an empty slot. It may or may not be assigned (provisioned).
Mismatch of Equipment and Attributes (MEA)	Incorrect equipment is installed. For example, the card plugged in is not compatible with the provisioned card or slot.
Automatic In-Service (AINS)	The entity is in a delay transition (to IS) state. The transition to IS is pending the correction of off-normal conditions on the entity.
MT (Maintenance)	The entity has been manually removed from service for maintenance. It is still capable of performing its provisioned functions. Traffic is still carried.

Table 2-9 Equipment Service State–Secondary (ONS 15454 SDH) (continued)

Secondary States (SST)	
Attribute Value	**Meaning**
DSBLD (Disabled)	The entity has been manually removed from service and cannot perform its provisioned functions. All of its provisioned services are disrupted. Traffic is not carried.
LPBK (Loopback)	The loopback command is in effect (any loopback type).

2.1.7.4 Topological Links

Topological links are unidirectional or bidirectional for ONS 15454 SDH NEs.

2.1.7.5 PTPs

All PTPs are bidirectional for the ONS 15454 SDH, except for PTPs on DWDM cards (optical amplifiers, optical multiplexers, optical demultiplexers, and optical add/drop multiplexers). The ports are always in channelized mode. PTPs support IN_SERVICE, OUT_OF_SERVICE, OUT_OF_SERVICE_MAINTENANCE, and AUTO_IN_SERVICE values for the service state attribute.

2.1.7.6 CTPs

All CTPs are in channelized mode for the ONS 15454 SDH. Alarm monitoring cannot be turned on or off for CTPs.

2.1.7.7 SNCs

One node supports a maximum of 192 bidirectional VC4 SNCs. The SNC name cannot exceed 48 characters. The NE enforces the SNC name character limit.

2.1.7.8 Equipment Protection

1:*N* equipment protection is supported for E1, E1-42, and DS3i cards. You must install protect cards (E1-*n*, E1-42, DS3i-*n*) in slot 3 or slot 15 on the same side of the shelf as the protected cards. Protect cards must match the cards they protect. For example, an E1-*n* card protects only E1 or E1-*n* cards. You must create the protection group. Installing E1-*n*, E1-42, or DS3i-*n* cards in slot 3 or slot 15 does not automatically protect other E1, E1-42, or DS3i cards. 1:*N* equipment protection is always revertive.

1:1 equipment protection is supported for E1, E1-42, E3, STM-1E-12, and DS3i cards. Install the protect card in an odd-numbered slot and install the working card in an adjacent even-numbered slot. For example, install the protect card in slot 1 and install the working card in slot 2.

2.1.7.9 Facility Protection

Facility protection is available for STM-*n* PTPs. Protect ports must match the working ports. For example, port 1 of an STM-1 (OC-3) card is protected only by port 1 on another STM-1 (OC-3) card.

2.1.7.10 Autodiscovery

ONS 15454 SDH autodiscovery is identical to ONS 15454 SONET autodiscovery, described in 2.1.8.10 Autodiscovery, page 2-14.

2.1.8 ONS 15454 SONET

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15454 SONET software releases that CTM R9.2 supports.

2.1.8.1 Shelves

All supported ONS 15454 SONET NEs have exactly one shelf each, with the exception of the ONS 15454 SONET MSTP R7.0, which can have up to eight shelves per NE.

2.1.8.2 Slots

Each ONS 15454 SONET shelf has 17 slots. The Timing Communications and Control Plus (TCC+) card must occupy slot 7 or slot 11. For more information, refer to the Cisco ONS 15454 SONET user documentation.

2.1.8.3 Equipment

All the cards in the ONS 15454 SONET R5.0 and later display administration and service states. For earlier software releases, these states are not applicable and CTM displays them as *N/A*. The following tables show the attribute values for these two states.

2.1.8.3.1 Equipment Administration State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. Table 2-10 lists the attribute values for the administration state. Table 2-11 and Table 2-12 list the attribute values for the service state.

Table 2-10 Equipment Administration State (ONS 15454 SONET)

Attribute Value	Meaning
In Service	Places the entity In Service.
Auto in Service	Places the entity in Auto in Service. The circuit is out of service until it receives a valid signal for the duration of the soak period, at which time the circuit state changes to In Service. During the soak period, alarms and loopbacks are suppressed but traffic is carried.
Out of Service–Maintenance	Removes the entity from service for maintenance. Alarms are no longer generated.
Out of Service	Removes the entity from service and disables it. Alarms are no longer generated and traffic is not carried.

2.1.8.3.2 Equipment Service State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. The attribute value for the service state is the same as that shown in CTM.

Table 2-11 Equipment Service State–Primary (ONS 15454 SONET)

Primary States (PST-PSTQ)	
Attribute Value	**Meaning**
In Service–Normal (IS_NR)	The entity is fully operational and performs as provisioned.
Out of Service–Management (OOS_MA)	The entity has been administratively removed from service.
Out of Service–Autonomous (OOS_AU)	The entity is not operational due to an autonomous event.
Out of Service–Autonomous and Management (OOS_AUMA)	The entity is not operational due to an autonomous event and has been administratively removed from service.

Table 2-12 Equipment Service State–Secondary (ONS 15454 SONET)

Secondary States (SST)	
Attribute Value	**Meaning**
UAS (Unassigned)	The entity is not provisioned in the database.
UEQ (Unequipped)	The entity has been removed, leaving an empty slot. It may or may not be assigned (provisioned).
MEA (Mismatch of Equipment and Attributes)	Incorrect equipment is installed. For example, the card plugged in is not compatible with the provisioned card or is not compatible with the slot.
AINS (Automatic In-Service)	The entity is in a delay transition (to IS) state. The transition to IS is pending the correction of off-normal conditions on the entity.
MT (Maintenance)	The entity has been manually removed from service for maintenance. It is still capable of performing its provisioned functions. Traffic is still carried.
DSBLD (Disabled)	The entity has been manually removed from service and cannot perform its provisioned functions. All of its provisioned services are disrupted. Traffic is not carried.
LPBK (Loopback)	The loopback command is in effect (any loopback type).

2.1.8.4 Topological Links

Topological links are unidirectional or bidirectional for ONS 15454 SONET NEs.

2.1.8.5 PTPs

All PTPs are bidirectional for the ONS 15454 SONET, except for PTPs on DWDM cards (optical amplifiers, optical multiplexers, optical demultiplexers, and optical add/drop multiplexers). The ports are always in channelized mode. PTPs support IN_SERVICE, OUT_OF_SERVICE, OUT_OF_SERVICE_MAINTENANCE, and AUTO_IN_SERVICE values for the service state attribute.

2.1.8.6 CTPs

All CTPs are in channelized mode for the ONS 15454 SONET. Alarm monitoring cannot be turned on or off for CTPs.

2.1.8.7 SNCs

One node supports a maximum of 144 bidirectional STS SNCs with an XC or Cross-Connect Virtual Tributary (XCVT) card installed or 576 bidirectional STS SNCs with a 10 Gigabit Cross-Connect (XC10G) card installed. If an XCVT or XC10G card is present, you can create a maximum of 336 bidirectional VT1.5 SNCs.

The SNC name cannot exceed 48 characters. The NE enforces the SNC name character limit.

2.1.8.8 Equipment Protection

1:*N* equipment protection is supported for DS1, DS3, DS3I, and DS3E cards. You must install protect cards (DS1N, DS3N, DS3IN, and DS3NE) in slot 3 or slot 15 on the same side of the shelf as the protected cards. Protect cards must match the cards they protect. For example, a DS1N protects only DS1 or DS1N cards. You must create the protection group. Installing DS1N or DS3N cards in slot 3 or slot 15 does not automatically protect other DS1 or DS3 cards. 1:*N* equipment protection is always revertive.

1:1 equipment protection is supported for DS1, DS3, DS3I, DS3E, EC-1, and DS3XM cards. Install the protect card in an odd-numbered slot and install the working card in an adjacent even-numbered slot. For example, install the protect card in slot 1 and install the working card in slot 2.

2.1.8.9 Facility Protection

Facility protection is available for OC-*n* PTPs. Protect ports must match the working ports. For example, port 1 on an OC-3 card is protected only by port 1 on another OC-3 card.

2.1.8.10 Autodiscovery

The ONS 15454 SONET supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all information, including the actual name, and an AVC event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.9 ONS 15530

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15530 software releases that CTM R9.2 supports.

2.1.9.1 Slots

The ONS 15530 is a single-shelf NE with 13 slots. It has a dual-CPU system with integrated switch fabric. The processor cards occupy slot 6 and slot 7. Slot 1 contains the multiplexer/demultiplexer motherboards, which are populated with optical multiplexer/demultiplexer modules. The remaining slots can contain:

- 10-port Enterprise System Connection (ESCON) module
- 1-port 10-Gigabit Ethernet (10GE) ITU grid dense wavelength division multiplexing (DWDM) module
- 1-port 10GE SR module
- 1-port transparent transponder module
- Optical Supervisory Channel (OSC) module consisting of two pluggable daughter modules for redundancy

2.1.9.2 PTPs

PTPs are both unidirectional and bidirectional for the ONS 15530. PTPs do not support the service state attribute.

The following interfaces are reported as PTPs:

- Transparent—Client-side transparent interface.
- Wavepatch—Passive optical interface on the transponder/trunk card that is patched to the filter interface on the multiplexer/demultiplexer card.
- Filter—Passive optical interface that passes a single wavelength on the multiplexer/demultiplexer card. It is patched to the wavepatch interface on the transponder trunk card.
- WDM—Passive external WDM interface on the multiplexer/demultiplexer card that carries multiple wavelengths, including the ones that have been added or dropped by the card. It can be used for internode or intermultiplexer/demultiplexer patching.
- Thru—Passive external WDM interface on the multiplexer/demultiplexer card that carries multiple wavelengths, including the ones that have been added but excluding the ones that have been dropped by the card. It can be used for internode or intermultiplexer/demultiplexer patching.
- Filterband—Passive special external WDM interface on the 16 wavelength multiplexer/demultiplexer card that passes through a special band of wavelengths. It is patched to the corresponding filter group interface on the other multiplexer/demultiplexer card.
- Filter group—Passive special external WDM interface on the 16 wavelength multiplexer/demultiplexer card that drops a special band of wavelengths.
- Wave—Optical lambda interface on the multiplexer/demultiplexer motherboard. Converts uncolored to one ITU grid wavelength.
- Oscfilter—Passive optical interface that passes OSC wavelengths on the multiplexer/demultiplexer card. It is patched to the wave interface on the transponder card.
- FastEthernet—Interface on the CPU card for Ethernet connectivity.
- EsconPhy—Physical ports on ESCON transceivers plugged into the ESCON line card.
- WaveEthernetPhy
- EthernetDcc
- TenGigEthernetPhy—Interface on the 10GE transceiver.

- voaFilterIn
- voaFilterOut
- voaBypassIn
- voaBypassOut
- voaIn
- voaOut

2.1.9.3 CTPs

CTM GateWay/CORBA does not support CTPs for the ONS 15530.

2.1.9.4 SNCs

CTM GateWay/CORBA does not support SNCs for the ONS 15530.

2.1.9.5 Equipment Protection

1:1 equipment protection is supported for the CPU card.

2.1.9.6 Equipment

The ONS 15530 reports all field-replaceable units (FRUs) as equipment. The chassis is also reported as equipment under the shelf equipment holder.

2.1.9.7 Topological Links

Topological links are unidirectional or bidirectional for ONS 15530 NEs. CTM reports both inter-NE and intra-NE links for the ONS 15530.

2.1.9.8 PTP Layer Rates

The ONS 15530 supports the following additional PTP layer rates:

- LR_Ten_Gigabit_Ethernet (TenGigabit Ethernet)
- LR_SYSPLEX_ISC_COMP (sysplexIscCompatibility)
- LR_SYSPLEX_ISC_PEER (sysplexIscPeer)
- LR_SYSPLEX_TIMER_ETR (sysplexTimerEtr)
- LR_SYSPLEX_TIMER_CLO (sysplexTimerClo)

2.1.10 ONS 15540

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15540 ESP and ESPx software releases that CTM R9.2 supports.

2.1.10.1 Slots

The ONS 15540 is a single-shelf NE with 13 slots. The ONS 15540 is a dual-CPU system where the processor cards occupy slot 7 and slot 8. Slot 1 and slot 2 contain the multiplexer/demultiplexer motherboards, which are populated with optical multiplexer/demultiplexer modules. The remaining slots contain the line card motherboards, which are populated with transponder modules.

2.1.10.2 PTPs

All PTPs are bidirectional for the ONS 15540. PTPs do not support the service state attribute.

Wavepatch interfaces on transponder cards and filter interfaces on multiplexer/demultiplexer cards are reported as PTPs, even though the cards are not visible externally from the fixed optical backplane. Wave interfaces on transponder cards are reported as PTPs even though they are internal interfaces, because alarms are generated on these interfaces and CTPs are not supported.

The following interfaces are reported as PTPs:

- Transparent—Client-side transparent interface.
- Wavepatch—Passive optical interface on the transponder card that is patched to the filter interface on the multiplexer/demultiplexer card.
- Filter—Passive optical interface that passes a single wavelength on the multiplexer/demultiplexer card. It is patched to the wavepatch interface on the transponder card.
- WDM—Passive external WDM interface on the multiplexer/demultiplexer card that carries multiple wavelengths, including the ones added or dropped by the card. It can be used for internode or intermultiplexer/demultiplexer patching.
- Thru—Passive external WDM interface on the multiplexer/demultiplexer card that carries multiple wavelengths, including the added ones but excluding the wavelengths dropped by the card. It can be used for internode or intermultiplexer/demultiplexer patching.
- Filterband—Passive special external WDM interface on the 16 wavelength multiplexer/demultiplexer card that passes through a special band of wavelengths. It is patched to the corresponding filter group interface on the other multiplexer/demultiplexer card.
- Filter group—Passive special external WDM interface on the 16 wavelength multiplexer/demultiplexer card that drops a special band of wavelengths.
- Wave—Optical lambda interface on the multiplexer/demultiplexer motherboard. Converts uncolored to one ITU grid wavelength.
- Oscfilter—Passive optical interface that passes OSC wavelengths on the multiplexer/demultiplexer card. It is patched to the wave interface on the transponder card.
- FastEthernet—Interface on the CPU card for Ethernet connectivity.
- TenGigEthernetPhy—Interface on the 10GE transceiver.
- WavePassThru—External interface on the POM dummy module for 10GE configurations.

2.1.10.3 CTPs

CTM GateWay/CORBA does not support CTPs for the ONS 15540.

2.1.10.4 SNCs

CTM GateWay/CORBA does not support SNCs for the ONS 15540.

2.1.10.5 Equipment Protection

1:1 equipment protection is supported for CPU cards.

2.1.10.6 Equipment

The ONS 15540 reports all FRUs as equipment. The chassis is also reported as equipment under the shelf equipment holder.

2.1.10.7 Topological Links

Topological links are unidirectional or bidirectional for ONS 15540 NEs. CTM reports both inter-NE and intra-NE links for the ONS 15540.

2.1.10.8 PTP Layer Rates

The ONS 15540 supports the following additional PTP layer rates:

- LR_Ten_Gigabit_Ethernet (TenGigabit Ethernet)
- LR_SYSPLEX_ISC_COMP (sysplexIscCompatibility)
- LR_SYSPLEX_ISC_PEER (sysplexIscPeer)
- LR_SYSPLEX_TIMER_ETR (sysplexTimerEtr)
- LR_SYSPLEX_TIMER_CLO (sysplexTimerClo)

2.1.11 ONS 15600 SONET

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15600 SONET software releases that CTM R9.2 supports.

2.1.11.1 Slots

The ONS 15600 SONET is a single-shelf NE with 14 slots. The Timing and Shelf Control (TSC) card must occupy slot 5 or slot 10. The Core Cross-Connect (CXC) cards are always 1+1 redundant in slots 6/7 and 8/9. Each CXC card occupies two slots. Slots 1 to 4 and 11 to 14 are reserved for optical I/O cards. Single Shelf Cross-Connect (SSXC) cards replace CXC cards from release 5.0 of the NE. However, due to a known caveat (CSCse87505), CTM and CTC display SSXC cards as CXC. For more information, refer to the Cisco ONS 15600 SONET user documentation.

2.1.11.2 Equipment

All cards in the ONS 15600 SONET R5.0 and later display administration and service states. For earlier software releases, these states are not applicable and CTM displays them as *N/A*.

2.1.11.2.1 Equipment Administration State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. Table 2-13 lists the attribute values for the administration state. Table 2-14 and Table 2-15 list the attribute values for the service state.

Table 2-13 Equipment Administration State (ONS 15600 SONET)

Attribute Value	Meaning
In Service	Places the entity In Service.
Auto in Service	Places the entity in Auto in Service. The circuit is out of service until it receives a valid signal for the duration of the soak period, at which time the circuit state changes to In Service. During the soak period, alarms and loopbacks are suppressed but traffic is carried.
Out of Service–Maintenance	Removes the entity from service for maintenance. Alarms are no longer generated.
Out of Service	Removes the entity from service and disables it. Alarms are no longer generated and traffic is not carried.

2.1.11.2.2 Equipment Service State

The attribute name is ACTUAL_EQUIPMENT_SERVICE_STATE and is displayed in getAdditionalInfo for the equipment. The attribute value for the service state is the same as that shown in CTM.

Table 2-14 Equipment Service State–Primary (ONS 15600 SONET)

Primary States (PST-PSTQ)	
Attribute Value	**Meaning**
In Service–Normal (IS_NR)	The entity is fully operational and performs as provisioned.
Out of Service–Management (OOS_MA)	The entity has been administratively removed from service.
Out of Service–Autonomous (OOS_AU)	The entity is not operational due to an autonomous event.
Out of Service–Autonomous and Management (OOS_AUMA)	The entity is not operational due to an autonomous event and has been administratively removed from service.

Table 2-15 Equipment Service State–Secondary (ONS 15600 SONET)

Secondary States (SST)	
Attribute Value	**Meaning**
Unassigned (UAS)	The entity is not provisioned in the database.
Unequipped (UEQ)	The entity has been removed, leaving an empty slot. It may or may not be assigned (provisioned).
Mismatch of Equipment and Attributes (MEA)	Incorrect equipment is installed. For example, the card plugged in is not compatible with the provisioned card or slot.
Automatic In-Service (AINS)	The entity is in a delay transition (to IS) state. The transition to IS is pending the correction of off-normal conditions on the entity.
Maintenance (MT)	The entity has been manually removed from service for maintenance. It is still capable of performing its provisioned functions. Traffic is still carried.

Table 2-15 Equipment Service State–Secondary (ONS 15600 SONET) (continued)

Secondary States (SST)	
Attribute Value	**Meaning**
Disabled (DSBLD)	The entity has been manually removed from service and cannot perform its provisioned functions. All of its provisioned services are disrupted. Traffic is not carried.
Loopback (LPBK)	The loopback command is in effect (any loopback type).

2.1.11.3 Topological Links

Topological links are unidirectional or bidirectional for ONS 15600 SONET NEs.

2.1.11.4 PTPs

All PTPs are bidirectional for the ONS 15600 SONET. These ports are always in channelized mode. PTPs support In Service and Out of Service–Maintenance values for the service state attribute.

2.1.11.5 CTPs

All CTPs are in channelized mode for the ONS 15600 SONET. Alarm monitoring cannot be turned on or off for CTPs.

2.1.11.6 SNCs

One ONS 15600 SONET NE supports a maximum of 1536 STS SNCs.

2.1.11.7 Protection

The ONS 15600 SONET supports 1+1 protection to create redundancy for optical ports. Protect ports must match the working ports. For example, port 1 on an OC-48 card can be protected only by port 1 on another OC-48 card.

2.1.11.8 Autodiscovery

The ONS 15600 SONET supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all information, including the actual name, and an AVC event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.12 ONS 15600 SDH

See the *Release Notes for Cisco Transport Manager Release 9.2* for the ONS 15600 SDH software releases that CTM R9.2 supports.

2.1.12.1 Slots

The ONS 15600 SDH is a single-shelf NE with 14 slots. The TSC card must occupy slot 5 or slot 10. The CXC cards are always 1+1 redundant in slots 6/7 and 8/9. Each CXC card occupies two slots. Slots 1 to 4 and 11 to 14 are reserved for optical I/O cards. SSXC cards replace CXC cards from release 5.0 of the NE. For more information, refer to the Cisco ONS 15600 SDH user documentation.

2.1.12.2 Topological Links

Topological links are unidirectional or bidirectional for ONS 15600 SDH NEs.

2.1.12.3 PTPs

All PTPs are bidirectional for the ONS 15600 SDH. These ports are always in channelized mode. PTPs support In Service and Out of Service–Maintenance values for the service state attribute.

2.1.12.4 CTPs

All CTPs are in channelized mode for the ONS 15600 SDH. Alarm monitoring cannot be turned on or off for CTPs.

2.1.12.5 SNCs

One ONS 15600 SDH NE supports a maximum of 1536 VC4 SNCs.

2.1.12.6 Protection

The ONS 15600 SDH supports 1+1 protection to create redundancy for optical ports. Protect ports must match the working ports. For example, port 1 on an STM-16 card can be protected only by port 1 on another STM-16 card.

2.1.12.7 Autodiscovery

The ONS 15600 SDH supports automatic discovery of connected elements. CTM can connect to one node and retrieve information about all connected nodes. The first time CTM connects to a node, it retrieves only the IP address of the connected node; CTM does not retrieve the node name. However, CTM must report this element to the users. Because the IP address is unique, CTM initializes the name of the newly discovered element with the IP address. An object creation event is generated for this managed element with the IP address as the name. Later, CTM connects to this element and retrieves all information, including the actual name, and an AVC event is generated for the managed element name. The NMS listens to the AVC event for the managed element name and invokes managedElementManager::ManagedElementManager_I:: getManagedElement.

2.1.13 MGX 8880/8850/8830

See the *Release Notes for Cisco Transport Manager Release 9.2* for the MGX software releases that CTM R9.2 supports.

2.1.13.1 Racks

The MGX 8880/8850/8830 is a single-rack NE.

2.1.13.2 Shelves

Each MGX 8880/8850/8830 NE has one shelf per rack.

2.1.13.3 Slots

The MGX 8880/8850 is a single-shelf NE with 32 slots. Slots 7 and 8 are reserved for the controller (PXM45 or PXM1E) front card. Slots 15/31 and 16/32 are reserved for the short reach (SR) ME front cards.

The MGX 8830 is a single-shelf NE with 14 slots. Slots 1 and 2 are reserved for the controller (PXM1E) front card. Slots 13/14 are reserved for SRME front cards.

2.1.13.4 Subslots

The MGX 8880/8850/8830 slot is divided into subslots or bays. The full-height cards (AXSM, VXSM, PXM, RPM, and PXM1E) have two subslots; the half-height cards (VISM, MPSM, and SRME) have one subslot. The subslots contain the back card for the corresponding front card.

2.1.13.5 PTPs

All PTPs are bidirectional for MGX 8880/8850/8830 NEs. PTPs do not support the service state attribute but do support the layer rate attribute.

2.1.14 MDS 9000

The MDS 9000 is a passive storage NE that does not contain any equipment. CTM reports PTPs for the MDS 9000 only if a topological link originates from or terminates on the MDS 9000. There are no alarms reported for the MDS 9000.

2.1.15 Unmanaged NEs

Unmanaged NEs are other vendor MEs in CTM. There is no equipment for these NEs. CTM reports PTPs for unmanaged NEs only if a topological link originates from or terminates on the unmanaged NE. There are no alarms reported for unmanaged NEs.

2.2 CTM-Specific Details

2.2.1 Layer Rate

Note The following table reports the layer rates supported by CTM. The layer rates are used for tasks such as link and circuit management, PM retrieval, port provisioning, and alarm handling. The Comments column contains further details on usage. Layer rates numbered 0 to 91 are TMF-defined, and layer rates numbered 92 and above are CTM-defined.

Table 2-16 Layer Rates Supported by CTM R9.2

Layer	Type	Layer Rate	Value	Comments
Physical	Gigabit	LR_Physical_2_5_Gigabit_ITU	108	OTU1 physical link
		LR_Physical_10_Gigabit_ITU	107	OTU2 physical link
		LR_Physical_40_Gigabit_ITU	152	OTU3 physical link
		LR_8b10b_2_5_Gigabit	121	—
	Ethernet	LR_Gigabit_Ethernet	68	—
		LR_Fast_Ethernet	61	—
		LR_Ten_Gigabit_Ethernet	106	WDM OCHCCs
		LR_Forty_Gigabit_Ethernet	162	—
		LR_DSR_Gigabit_Ethernet	87	Physical links
		LR_LEX	164	—
	SONET/SDH	LR_DSR_OC1_STM0	72	—
		LR_DSR_OC3_STM1	73	—
		LR_DSR_OC12_STM4	74	WDM OCHCCs
		LR_DSR_OC24_STM8	75	—
		LR_DSR_OC48_STM16	76	Physical SONET/SDH links
		LR_DSR_OC192_STM64	77	—
		LR_DSR_OC768_STM256	78	—
	FICON	LR_FICON_1G	112	—
		LR_FICON_2G	113	—
		LR_FICON_4G	137	—
	Fiber Channel	LR_FC_12_133M	62	120M FC
		LR_FC_25_266M	63	250M FC
		LR_FC_50_531M	64	500M FC
		LR_FC_100_1063M	65	1G FC
		LR_FC_200_2125M	111	2G FC links
		LR_FC_400_4250M	134	4G FC
		LR_FC_800_8500M	160	8G FC
		LR_FC_1000_10518M	133	10G FC
	Optical	LR_PHYSICAL_OPTICAL	47	—
	Electrical	LR_PHYSICAL_ELECTRICAL	46	—
Optical Transmission Section	OTS	LR_Optical_Transmission_Section	42	OTS links

Table 2-16 Layer Rates Supported by CTM R9.2 (continued)

Layer	Type	Layer Rate	Value	Comments
Optical Multiplex Section	OMS	LR_Optical_Multiplex_Section	41	—
		LR_OMS_DWDM_18_CH	125	OMS links
		LR_OMS_DWDM_18_CH	126	—
Optical Channel	OCH	LR_Optical_Channel	40	For WDM wavelength
		LR_OCH_2_5_Gigabit_ITU	110	OCH links
		LR_OCH_10_Gigabit_ITU	109	—
		LR_OCH_40_Gigabit_ITU	163	—
		LR_OCH_Transport_Unit	10001	For OCH Trail Tunnel SNCs
Optical Transport	Path, Section, FEC	LR_Optical_Transport_Path	116	—
		LR_Optical_Transport_Section	117	—
		LR_Optical_Transport_FEC	118	—
SONET/SDH	OC/STM Line/Section	LR_Line_OC1_STS1_and_MS_STM0	24	—
		LR_Line_OC3_STS3_and_ MS_STM1	25	—
		LR_Line_OC12_STS12_and_ MS_STM4	26	—
		LR_Line_OC24_STS24_and_MS_STM8	89	—
		LR_Line_OC48_STS48_and_ MS_STM16	27	—
		LR_Line_OC192_STS192_and_ MS_STM64	28	—
		LR_Line_OC768_STS768_and_MS_STM256	91	—
		LR_Section_OC1_STS1_and_RS_STM0	19	—
		LR_Section_OC3_STS3_and_RS_STM1	20	—
		LR_Section_OC12_STS12_and_RS_STM4	21	—
		LR_Section_OC24_STS24_and_RS_STM8	88	SONET/SDH links
		LR_Section_OC48_STS48_and_RS_STM16	22	—
		LR_Section_OC192_STS192_and_RS_STM64	23	—
		LR_Section_OC768_STS768_and_RS_STM256	90	—
		LR_SONET_SDH_Composite	10002	For special links created automatically by CTC and used for SONET/SDH circuit routing through DWDM transponders.

Table 2-16 Layer Rates Supported by CTM R9.2 (continued)

Layer	Type	Layer Rate	Value	Comments
	CCAT VCAT	LR_STS1_and_AU3_High_Order_VC3	14	Only SONET OC cards
	SNC	LR_Low_Order_TU3_VC3	13	Only SDH STM cards
		LR_STS3c_and_AU4_VC4	15	—
		LR_STS12c_and_VC4_4c	16	Server Trail links
		LR_STS6c_and_VC4_2c	98	SONET/SDH SNCs
		LR_STS9c_and_VC4_3c	99	—
		LR_STS192c_and_VC4_64c	161	—
		LR_VT1_5_and_TU11_VC11	10	—
		LR_VT2_and_TU12_VC12	11	—
		LR_VT6_and_TU2_VC2	12	—
Layer 2	L2	LR_L2_10_Gigabit_Ethernet	153	PPC L2 links
		LR_L2_1_Gigabit_Ethernet	154	—
		LR_Not_Applicable	1	ML/CE L2 topologies
		LR_POS_100M	119	—
		LR_POS_Gigabit	120	—
		LR_COS_100M	122	ML/CE PM retrieval
		LR_COS_Gigabit	123	—
		LR_RPR_IEEE_802_17	150	—
PDH	T*, E*	LR_T1_and_DS1_1_5M	2	—
		LR_T2_and_DS2_6M	3	—
		LR_T3_and_DS3_45M	4	—
		LR_E1_2M	5	—
		LR_E2_8M	6	—
		LR_E3_34M	7	—
		LR_E4_140M	8	—
		LR_E5_565M	9	—
ATM	VC, VP, UNI/NNI	LR_ATM_NI	43	For ATM Network Interfaces
		LR_ATM_VP	44	For ATM Virtual Paths
		LR_ATM_VC	45	For ATM Virtual Channels
Video	D1, HDTV, DV	LR_D1_Video	58	Video-capable port
		LR_HDTV	146	—
		LR_DV_6000	148	—
		LR_DVBASI	156	—

Table 2-16 Layer Rates Supported by CTM R9.2 (continued)

Layer	Type	Layer Rate	Value	Comments
Data/Storage	Escon, ETR, ISC, IB, SYSPLEX	LR_ESCON	59	IBM protocol for Mainframe
		LR_ETR	60	IBM protocol for Mainframe
		LR_ISC3_PEER_1G	143	—
		LR_ISC3_PEER_2G	144	—
		LR_ISC1	155	—
		LR_IB	157	—
		LR_SYSPLEX_ISC_COMP	102	Infiniband
		LR_SYSPLEX_ISC_PEER	103	—
		LR_SYSPLEX_TIMER_ETR	104	—
		LR_SYSPLEX_TIMER_CLO	105	—
		LR_PASS_THROUGH	149	—

2.2.2 Inventory

The ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, ONS 15530, ONS 15540, ONS 15600 SONET, and ONS 15600 SDH are single-shelf NEs. However, in future they may be expanded to multishelf, multibay NEs. Considering this, CTM reports the following:

- Rack = 1 and shelf = 1 for ONS 15310, ONS 15327, ONS 15530, ONS 15540, ONS 15600 SONET, and ONS 15600 SDH inventory. This implementation allows CTM to accommodate future changes quickly.
- ONS 15454 MSTP NEs support up to eight shelves per NE. The shelf ID identifies a specific shelf on the NE. The rack ID remains 1 for all shelves.
- For the ONS 15454 SDH, the EFCA shelf and the main shelf are considered to be a single logical shelf with slots numbering from 1 to 29.
- Rack = 1 and shelf = 1 for ONS 15216 NEs.
- For the ONS 15216 DCU, CTM reports the DCU chassis as the shelf. CTM reports two slots as equipment holders contained by the shelf. Any DCM module is reported as equipment under the slot.
- For the ONS 15216 FlexLayer, CTM reports the FlexLayer chassis as the shelf. CTM reports four slots as equipment holders contained by the shelf. Any provisioned module is reported as equipment under the slot.
- For the remaining ONS 15216 NEs, CTM reports only one type of equipment under the shelf. The equipment name is the same as the NE name.
- For some supported NEs, equipment does not fit in the normal equipment holders (rack, shelf, slot, and so on) and does not have provisioning functions. For inventory purposes, CTM defines a special EquipmentHolder named AdditionalHolder to hold this equipment.
- For the ONS 15454 SONET R3.4 and later, the Alarm Interface Extension (AIE) is reported in AdditionalHolder 1 (if installed). The Alarm Extension Panel (AEP) is reported in AdditionalHolder 2.

- For the ONS 15454 SDH R3.4 and later, the AIE (if installed) is reported in AdditionalHolder 1.
- For the ONS 15600 SONET R5.0 and later, pluggable port modules (PPMs) and pluggable I/O modules (PIMs) on ASAP cards are reported as subslots and port holders, respectively.
- For the ONS 15600 SDH R8.0, PPMs and PIMs on ASAP cards are reported as subslots and port holders, respectively.
- For unmanaged NEs, CTM reports only ManagedElement. CTM does not report any equipment holder or equipment.
- For storage NEs, CTM reports only ManagedElement. CTM does not report any equipment holder or equipment.

MGX 8880/8850/8830 NEs are single-shelf NEs. However, in the future, they may be expanded to multishelf, multibay NEs. Considering this, CTM reports the following:

- Rack = 1 and shelf = 1 for the MGX 8880/8850/8830 inventory. This implementation allows CTM to accommodate future changes quickly.
- CTM reports MGX slots and peripheral holders as equipment holders contained by the shelf. Any provisioned card is reported as equipment under the slot, and any provisioned peripheral is reported as equipment under the peripheral holder.
- CTM reports MGX slots with a provisioned card as equipment holders. CTM does not report empty slots as equipment holders.
- For M2 NEs, CTM reports the slots with a provisioned card as equipment holders; this is applicable only to slots 1 to 3. Other equipment is not managed.
- For M6 NEs, CTM reports the slots with a provisioned card as equipment holders; this is applicable only to slots 1 to 8. Other equipment is not managed.

2.2.3 Interfaces

Unless specified otherwise, all get*XXX* methods are synchronous.

CTM supports all of the ONS 15xxx-series NEs. However, methods supported for one NE type might not be supported for another NE type. In this case, CTM returns a NOT_IMPLEMENTED ProcessingFailureException for the unsupported method.

Note CTM supports TMF-defined and Cisco-proprietary application programming interfaces (APIs) and data structures. The Cisco-proprietary APIs and data structures are subject to change in future releases of CTM to accommodate the introduction of new features or to maintain TMF compliance. This is especially true if the TMF were to define the same method with a different behavior.

In such cases, the affected API name is changed without changing the behavior. The renamed API retains the signature and behavior of the original API.

A standard deprecation period of one CTM release is allowed on the obsolete APIs and data structure. However, Cisco is willing to consider exceptions that minimize the business impact to customers.

2.2.4 Object Details

2.2.4.1 Multilayer Subnetwork

A subnetwork is a collection of managed elements that reflects network connectivity, in most cases. There are two ways to create a new subnetwork through the CTM client:

- In the Domain Explorer window, choose **File > Add Network Element(s)**. Choose **<SELF>** as the GNE ID and **<System Default>** as the subnetwork ID. CTM creates a new subnetwork and assigns a name to the new subnetwork object. The naming convention is Subnetwork-*ID*, where *ID* = 00000001, 00000002, *n*.
- Create a subnetwork object from the CTM client by choosing **File > Subnetwork Explorer**. In the Subnetwork Explorer window, select a network partition and choose **File > Add New Subnetwork**. Enter the subnetwork name.

There are four ways to add NEs to subnetworks:

- Use the CTM client to specify a subnetwork for the new NE. If the new NE is a GNE and you choose **<System Default>** as the subnetwork ID, the new subnetwork is created as described in the first bullet above.
- If the new NE is not a GNE, choose an existing subnetwork as the subnetwork ID. CTM adds this NE to the subnetwork.
- If the new NE is connected to an existing NE on the network, CTM discovers the NE and automatically adds it to the existing NE's subnetwork. No user action is required.
- Use the CTM client to move the NE from one subnetwork to another subnetwork.

SNC creation is limited within a multilayer subnetwork scope. For example, assume Subnetwork-1 contains two NEs, A and B, that are connected to each other with an autodiscovered link. Subnetwork-2 contains two NEs, C and D, that are connected to each other with an autodiscovered link. There is also a manually routable link between NEs B and C. To create an SNC between NEs A and D, you must move NEs C and D into Subnetwork-1, or move NEs A and B into Subnetwork-2.

CTM includes a feature to automatically group NEs in subnetworks. This feature is enabled by default. If a routable link is created between two NEs in different subnetworks and CTM merges the two subnetworks, the result is reflected in CTM GateWay/CORBA. The two subnetworks are merged automatically when the Automatically Group NEs in Subnetworks feature is enabled in CTM.

2.2.4.2 SNCs

SNCs are supported only for ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH NEs. CTM does not support SNC in pending state. However, CTM SNC has a proprietary deleting state.

2.2.4.3 AVC Events

In certain situations, CTM cannot provide the list of changed object attributes. In this case, CTM generates an AVC event with an empty attribute list. The NMS makes a get call to retrieve the object details.

2.2.4.4 Managed Element and Object Creation Event

You can use the CTM client to add new managed elements. If the name of the new element is incorrect, an object creation event is generated with the incorrect name. Later, when CTM connects to the actual NE, it retrieves the correct name. CTM generates an AVC event to notify the NMS of the correct name.

2.2.5 Naming Conventions

The following sections describe naming conventions for objects that CTM reports.

2.2.5.1 Multilayer Subnetwork

You define the name in the CTM client. If you do not specify a name, CTM assigns a default name.

2.2.5.2 Managed Element

The ME reports the name.

2.2.5.3 Physical Termination Point

The PTP name represents the position of the PTP with respect to the equipment. For example, port number 4 on equipment in slot 15 is reported as /rack=1/shelf=1/slot=15/port=4.

For multishelf NEs, the rack and shelf number must be included in the PTP name. ONS 15454 MSTP NEs support up to eight shelves per NE. The shelf ID identifies a specific shelf on the NE. The rack ID remains 1 for all shelves.

This PTP naming convention is used to represent the front Ethernet ports for the ONS 15454 SONET and ONS 15454 SDH ML-series cards. The back Ethernet ports are reported as floating termination points (FTPs).

For class of service (CoS) data on data cards, the PTP name specifies the interface (POS, Fast Ethernet, Gigabit Ethernet), direction (input/output), and CoS level (0 to 7).

For example, CoS data with a POS interface, input direction, and CoS level 3 for slot 1, port 1 is reported as rack=1/shelf=1/slot=1/port=1:POS/Input/3.

Similarly, CoS data with a Fast Ethernet interface, output direction, and CoS level 5 for slot 1, port 1 is reported as /rack=1/shelf=1/slot=1/port=1:FastEthernet/Output/5.

For the ONS 15600 SONET, the ASAP card on slot 3, port 1 on PPM 3 and PIM 2 is reported as /rack=1/shelf=1/slot=3/sub_slot=2/ppm_holder=3/port=1.

For the ONS 15310 CL, ONS 15310 MA SONET, and ONS 15310 MA SDH, a port on PPM number 2 is reported as /rack=1/shelf=1/slot=2/ppm_holder=2/port=1.

For the ONS 15454 SONET and ONS 15454 SDH, a port on PPM number 2 is reported as /rack=1/shelf=1/slot=2/ppm_holder=2/port=1.

For the ONS 15530 and ONS 15540, the PTP name represents the position of the PTP with respect to the equipment. It also specifies the CLI name, since there can be more than one interface type with the same physical position. Numbering for equipmentHolders and ports starts from 0 in the NE. However, the TMF convention is to start numbering from 1 for equipmentHolders and ports. Therefore, CTM translates PTP names.

For example, the wavepatch0/0/0 port (which is present in slot 0, subslot 0 in the NE) is reported as /rack=1/shelf=1/slot=1/subslot=1/port=1:wavepatch0/0/0.

The transparent0/0/0 port (which is present in slot 0, subslot 0 in the NE) is reported as /rack=1/shelf=1/slot=1/subslot=1/port=1:transparent0/0/0.

Some PTPs, such as wdm0/0, do not have an associated port number. In these cases, the port number is set to 1.

Wdm0/0 (which is present in slot 0 and subslot 0 in the NE) is reported as /rack=1/shelf=1/slot=1/subslot=1/port=1:wdm0/0.

Wave0 interface (which is present in slot 0 in the NE) is reported as /rack=1/shelf=1/slot=1/port=1:wave0. Because the subslot does not apply to this port, it is skipped.

For unmanaged NEs, the PTP native name is a free-format string.

For the MGX 8880/8850/8830, the PTP name represents the position of the PTP with respect to the equipment. For example, port number 4 on equipment in slot 10 and bay 1 is reported as /rack=1/shelf=1/slot=10/subslot=1/port=4.

2.2.5.4 Connection Termination Point

Naming conventions for SONET, SDH, and plesiochronous digital hierarchy (PDH) CTP are defined by TMF.814 in a document named *objectNaming.html*. CTM follows these rules.

For OC-*n* or STM-*n* cards on the ONS 15327 R3.3 or later, ONS 15454 SONET R3.2.1 or later, and ONS 15454 SDH R3.3 or later, CTM supports STS6c, STS9c, and STS24c layer rates. On the ONS 15327 R5.0 or later, ONS 15454 SONET R5.0 or later, and ONS 15454 SDH R5.0 or later, CTM supports STS18c and STS36c layer rates. These are not conventional SONET/SDH rates; therefore, the TMF document does not cover their naming.

The following table shows the CTM CTPs for STS6c, STS9c, STS24c, STS18c, and STS36c layer rates. The number after '-r' is the difference between the STS/VC4 number and the previous edge STS/VC4 number.

Table 2-17 CTPs for STS6c, STS9c, STS18c, STS24c, and STS36c

Layer Rate	CTP Tuple	Comments
PTP: OC12		
sts6c_vc4_2c	/sts6c_vc4_2c=1, 2	CTP is on the edge of sts6c, STS#1 and STS#7.
	/sts6c_vc4_2c=1-r=[1..5]	CTP is not on the edge of sts6c. For example, if CTP is on STS#4, the name is /sts6c_vc4_2c=1-r=3.
sts9c_vc4_3c	/sts9c_vc4_3c=1	CTP is on the edge of sts9c, STS#1.
	/sts9c_vc4_3c=1-r=[1..3]	CTP is not on the edge of sts9c. For example, if CTP is on STS#4, the name is /sts9c_vc4_3c=1-r=3.
PTP: OC48		
sts6c_vc4_2c	/sts6c_vc4_2c=[1..8]	CTP is on the edge of sts6c, STS#1, STS#7… STS#43.
	/sts6c_vc4_2c=[1..7]-r=[1..5]	CTP is not on the edge of sts6c. For example, if CTP is on STS#10, the name is /sts6c_vc4_2c=2-r=3.
sts9c_vc4_3c	/sts9c_vc4_3c=[1..5]	CTP is on the edge of sts9c, STS#1, STS#10… STS#37.
	/sts9c_vc4_3c=[1..5]-r=[1..8]	CTP is not on the edge of sts9c. For example, if CTP is on STS#16, the name is /sts9c_vc4_3c=2-r=6.

Table 2-17 CTPs for STS6c, STS9c, STS18c, STS24c, and STS36c (continued)

Layer Rate	CTP Tuple	Comments
sts24c_vc4_8c	/sts24c_vc4_8c=1, 2	CTP is on the edge of sts24c, STS#1, STS#25.
	/sts24c_vc4_8c=1-r=[1..23]	CTP is not on the edge of sts24c. For example, if CTP is on STS#10, the name is /sts24c_vc4_8c=1-r=9.
sts18c_vc4_6c	/sts18c_vc4_6c=1, 2	CTP is on the edge of sts18c, STS#1, STS#19.
	/sts18c_vc4_6c=1-r=[1..17]	CTP is not on the edge of sts18c. For example, if CTP is on STS#10, the name is /sts18c_vc4_6c=1-r=9.
sts36c_vc4_12c	/sts36c_vc4_12c=1	CTP is on the edge of sts36c, STS#1.
	/sts36c_vc4_12c=1-r=[1..35]	CTP is not on the edge of sts36c. For example, if CTP is on STS#10, the name is /sts36c_vc4_12c=1-r=9.
PTP: OC192		
sts6c_vc4_2c	/sts6c_vc4_2c=[1..32]	CTP is on the edge of sts6c, STS#1, STS#7… STS#187.
	/sts6c_vc4_2c=[1..31]-r=[1..5]	CTP is not on the edge of sts6c. For example, if CTP is on STS#4, the name is /sts6c_vc4_2c=1-r=3.
sts9c_vc4_3c	/sts9c_vc4_3c=[1..21]	CTP is on the edge of sts9c, STS#1, STS#10… STS#181.
	/sts9c_vc4_3c=[1..20]-r=[1..8]	CTP is not on the edge of sts9c. For example, if CTP is on STS#52, the name is /sts9c_vc4_3c=6-r=6.
sts18c_vc4_6c	/sts18c_vc4_6c=1,10	CTP is on the edge of sts18c, STS#1, STS#19.
	/sts18c_vc4_6c=1-r=[1..17]	CTP is not on the edge of sts18c. For example, if CTP is on STS#10, the name is /sts18c_vc4_6c=1-r=9.
sts24c_vc4_8c	/sts24c_vc4_8c=[1..8]	CTP is on the edge of sts24c, STS#1, STS#25… STS#169.
	/sts24c_vc4_8c=[1..7]-r=[1..23]	CTP is not on the edge of sts24c. For example, if CTP is on STS#61, the name is /sts24c_vc4_8c=3-r=12.
sts36c_vc4_12c	/sts36c_vc4_12c=1,5	CTP is on the edge of sts36c, STS#1, STS#37.
	/sts36c_vc4_12c=1-r=[1..35]	CTP is not on the edge of sts36c. For example, if CTP is on STS#10, the name is /sts36c_vc4_12c=1-r=9.
PTP: STM4		
sts6c_vc4_2c	/sts6c_vc4_2c=1,2	CTP is on the edge of vc4_2c, VC4#1, and VC4#3.
	/sts6c_vc4_2c=1-r=1	CTP is on VC4#2.
sts9c_vc4_3c	/sts9c_vc4_3c=1	CTP is on the edge of vc4_3c, VC4#1.
	/sts9c_vc4_3c=1-r=1	CTP is on VC4#2.
PTP: STM16		
sts6c_vc4_2c	/sts6c_vc4_2c=[1..8]	CTP is on the edge of vc4_2c, VC4#1, VC4#3… VC4#15.
	/sts6c_vc4_2c=[1..7]-r=1	CTP is not on the edge of vc4_2c. For example, if CTP is on VC4#4, the name is /sts6c_vc4_2c=2-r=1.
sts9c_vc4_3c	/sts9c_vc4_3c=[1..5]	CTP is on the edge of vc4_3c, VC4#1, VC4#4… VC4#13.
	/sts9c_vc4_3c=[1..5]-r=[1..2]	CTP is not on the edge of vc4_3c. For example, if CTP is on VC4#6, the name is /sts9c_vc4_3c=2-r=2.

Table 2-17 CTPs for STS6c, STS9c, STS18c, STS24c, and STS36c (continued)

Layer Rate	CTP Tuple	Comments
sts18c_vc4_6c	/sts18c_vc4_6c=1,2	CTP is on the edge of vc4_6c, VC4#1, VC4#7.
	/sts18c_vc4_6c=1-r=[1..5]	CTP is not on the edge of vc4_6c. For example, if CTP is on VC4#3, the name is /sts18c_vc4_6c=1-r=2.
sts24c_vc4_8c	/sts24c_vc4_8c=1,2	CTP is on the edge of vc4_8c, VC4#1, VC4#9.
	/sts24c_vc4_8c=1-r=[1..7]	CTP is not on the edge of vc4_8c. For example, if CTP is on VC4#4, the name is /sts24c_vc4_8c=1-r=3.
sts36c_vc4_12c	/sts36c_vc4_12c=1	CTP is on the edge of vc4_12c, VC4#1.
	/sts36c_vc4_12c=1-r=[1..11]	CTP is not on the edge of vc4_12c. For example, if CTP is on VC4#10, the name is /sts36c_vc4_12c=1-r=9.
PTP: STM64		
sts6c_vc4_2c	/sts6c_vc4_2c=[1..32]	CTP is on the edge of vc4_2c, VC4#1, VC4#3… VC4#63.
	/sts6c_vc4_2c=[1..31]-r=1	CTP is not on the edge of vc4_2c. For example, if CTP is on VC4#20, the name is /sts6c_vc4_2c=10-r=1.
sts9c_vc4_3c	/sts9c_vc4_3c=[1..21]	CTP is on the edge of vc4_3c, VC4#1, VC4#4… VC4#61.
	/sts9c_vc4_3c=[1..21]-r=[1..3]	CTP is not on the edge of vc4_3c. For example, if CTP is on VC4#6, the name is /sts9c_vc4_3c=2-r=2.
sts18c_vc4_6c	/sts18c_vc4_6c=1,9	CTP is on the edge of vc4_6c, VC4#1, VC4#7.
	/sts18c_vc4_6c=1-r=[1..5]	CTP is not on the edge of vc4_6c. For example, if CTP is on VC4#3, the name is /sts18c_vc4_6c=1-r=2.
sts24c_vc4_8c	/sts24c_vc4_8c=[1..8]	CTP is on the edge of vc4_8c, VC4#1, VC4#9… VC4#57.
	/sts24c_vc4_8c=[1..7]-r=[1..7]	CTP is not on the edge of vc4_8c. For example, if CTP is on VC4#28, the name is /sts24c_vc4_8c=4-r=3.
sts36c_vc4_12c	/sts36c_vc4_12c=1,4	CTP is on the edge of vc4_12c, VC4#1.
	/sts36c_vc4_12c=1-r=[1..11]	CTP is not on the edge of vc4_12c. For example, if CTP is on VC4#10, the name is /sts36c_vc4_12c=1-r=9.

In addition, the following CTP name support has been added:

PTP	Layer Rate	CTP Tuple
STM1	vt2_tu12	/sts3c_au4-*j*=1/vt2_tu12-*k*=[1..3]-*l*=[1..7]-*m*=[1..3] The qualifier strings represent the following values: • *j*—AUG index • *k*—TUG-3 or AU-3 index • *l*—TUG-2 index • *m*—TU-12 or TU-11 index

For CTPs used for WDM SNCs (optical channel network connection [OCHNC], optical channel client connection [OCHCC], and OCH trail tunnels), use the naming conventions defined in TMF: /frequency=*nnn.mm*, where *nnn.mm* is a decimal representing the frequency in terahertz (THz).

For unidirectional CTP, the naming convention is to add "/direction=src or sink" in front of the name. Because the CTPs in WDM SNC are all unidirectional, the name is "/direction=src or sink/frequency=*nnn.mm*" for WDM SNC CTPs.

Because the CTM client uses wavelengths to represent an optical channel, enter the corresponding wavelength value in the nativeEMSName field for the CTP.

The formula for converting wavelength to frequency is:

frequency (THz) = 299792458 / wavelength (nm) / 1000

The formula for converting frequency back to wavelength is:

wavelength (nm) = 299792458 / frequency (THz) / 1000

The possible values of the frequency *nnn.mm* (in THz) in the CTP and the corresponding wavelength values are:

For OCHNC:

- C-band

195.90(1530.33), 195.80(1531.12), 195.70(1531.90), 195.60(1532.68), 195.40(1534.25), 195.30(1535.04), 195.20(1535.82), 195.10(1536.61), 194.90(1538.19), 194.80(1538.98), 194.70(1539.77), 194.60(1540.56), 194.40(1542.14), 194.30(1542.94), 194.20(1543.73), 194.10(1544.53), 193.90(1546.12), 193.80(1546.92), 193.70(1547.72), 193.60(1548.51), 193.40(1550.12), 193.30(1550.92), 193.20(1551.72), 193.10(1552.52), 192.90(1554.13), 192.80(1554.94), 192.70(1555.75), 192.60(1556.55), 192.40(1558.17), 192.30(1558.98), 192.20(1559.79), 192.10(1560.61).

- L-band

1577.86, 1578.69, 1579.52, 1580.35, 1581.18, 1582.02, 1582.85, 1583.69, 1584.53, 1585.36, 1586.20, 1587.04, 1587.88, 1588.73, 1589.57, 1590.41, 1591.26, 1592.10, 1592.95, 1593.79, 1594.64, 1595.49, 1596.34, 1597.19, 1598.04, 1598.89, 1599.75, 1600.60, 1601.46, 1602.31, 1603.17, 1604.03.

For OCHCC:

- C-band (odd)

1529.55 nm, 1530.33 nm, 1531.12 nm, 1531.90 nm, 1532.68 nm, 1533.47 nm, 1534.25 nm, 1535.04 nm, 1535.82 nm, 1536.61 nm, 1537.40 nm, 1538.19 nm, 1538.98 nm, 1539.77 nm, 1540.56 nm, 1541.35 nm, 1542.14 nm, 1542.94 nm, 1543.73 nm, 1544.53 nm, 1545.32 nm, 1546.12 nm, 1546.92 nm, 1547.72 nm, 1548.51 nm, 1549.32 nm, 1550.12 nm, 1550.92 nm, 1551.72 nm, 1552.52 nm, 1553.33 nm, 1554.13 nm, 1554.94 nm, 1555.75 nm, 1556.55 nm, 1557.36 nm, 1558.17 nm, 1558.98 nm, 1559.79 nm, 1560.61 nm, 1561.42 nm.

- C-band (even)

1529.94 nm, 1530.73 nm, 1531.51 nm, 1532.29 nm, 1533.07 nm, 1533.86 nm, 1534.64 nm, 1535.43 nm, 1536.22 nm, 1537.00 nm, 1537.79 nm, 1538.58 nm, 1539.37 nm, 1540.16 nm, 1540.95 nm, 1541.75 nm, 1542.54 nm, 1543.33 nm, 1544.13 nm, 1544.92 nm, 1545.72 nm, 1546.52 nm, 1547.32 nm, 1548.12 nm, 1548.92 nm, 1549.71 nm, 1550.52 nm, 1551.32 nm, 1552.12 nm, 1552.93 nm, 1553.73 nm, 1554.54 nm, 1555.34 nm, 1556.15 nm, 1556.96 nm, 1557.77 nm, 1558.58 nm, 1559.39 nm, 1560.20 nm, 1561.01 nm, 1561.83 nm.

- L-band (odd)

1570.83 nm, 1571.65 nm, 1572.48 nm, 1573.30 nm, 1574.13 nm, 1574.95 nm, 1575.78 nm, 1576.61 nm, 1577.44 nm, 1578.27 nm, 1579.10 nm, 1579.93 nm, 1580.77 nm, 1581.60 nm, 1582.44 nm, 1583.27 nm, 1584.11 nm, 1584.95 nm, 1585.78 nm, 1586.62 nm, 1587.46 nm, 1588.30 nm, 1589.15 nm, 1589.99 nm, 1590.83 nm, 1591.68 nm, 1592.52 nm, 1593.37 nm, 1594.22 nm, 1595.06 nm, 1595.91 nm, 1596.76 nm, 1597.62 nm, 1598.47 nm, 1599.32 nm, 1600.17 nm, 1601.03 nm, 1601.88 nm, 1602.74 nm, 1603.60 nm.

- L-band (even)

1571.24 nm, 1572.06 nm, 1572.89 nm, 1573.71 nm, 1574.54 nm, 1575.37 nm, 1576.20 nm, 1577.03 nm, 1577.86 nm, 1578.69 nm, 1579.52 nm, 1580.35 nm, 1581.18 nm, 1582.02 nm, 1582.85 nm, 1583.69 nm, 1584.53 nm, 1585.36 nm, 1586.20 nm, 1587.04 nm, 1587.88 nm, 1588.73 nm, 1589.57 nm, 1590.41 nm, 1591.26 nm, 1592.10 nm, 1592.95 nm, 1593.79 nm, 1594.64 nm, 1595.49 nm, 1596.34 nm, 1597.19 nm, 1598.04 nm, 1598.89 nm, 1599.75 nm, 1600.60 nm, 1601.46 nm, 1602.31 nm, 1603.17 nm, 1604.03 nm.

2.2.5.5 Floating Termination Point

The FTP naming contains three tuples. The first tuple is the EMS name, the second tuple is the ME name, and the third tuple is the FTP name. The value of the name field in the third tuple is "FTP." The native FTP name is a free-format string. However, CTM GateWay/CORBA follows the convention for PTP and CTP, if applicable.

All Ethernet termination points (TPs) for circuit provisioning on E-series, G-series, ML-series, CE-100T-8, and FCMR cards are modeled as FTPs. The convention for the FTP values for Ethernet TPs is as follows:

- For FTPs used in single-card EtherSwitch, CTM uses /rack=1/shelf=1/slot=*n*/EtherSwitchGroup.
- For FTPs used in multicard EtherSwitch, CTM uses /MulticardEtherSwitchGroup.
- For FTPs used in linear-card EtherSwitch or G-series cards, CTM uses /rack=1/shelf=1/slot=*n*/port=*m*.
- For FTPs used in ML-series cards, CTM uses /rack=1/shelf=1/slot=*n*/port=*m*, where *m* = 0,1.
- To create or report VT tunnel, CTM uses /VTTunnel.
- To create or report VC tunnel, CTM uses /VCTunnel.
- To create VC tunnel for VC3 port grouping and SNCs, CTM uses /rack=1/shelf=1/slot=*n*/vc4=*m*, where *n* is from 1 to 17 and *m* is from 1 to 4.
- For VT aggregation point circuits used to create or report VT aggregation point (VAP) circuits, CTM uses /VAPTP for SONET NEs.
- For low-order aggregation point circuits, CTM uses /LAPTP for SDH NEs.

2.2.5.6 Equipment

For a list of equipment that CTM GateWay/CORBA reports for each NE type, see Appendix D, "Equipment List for NEs."

2.2.5.7 Equipment Holder

The TMF defines the EquipmentHolder naming convention. Numbering is done from left to right and from top to bottom. For example, slot 15 is named /rack=1/shelf=1/slot=15.

For ONS 15xxx NEs, the numbering for EquipmentHolders starts from 0 in the NE. The TMF convention is to start numbering from 1 for EquipmentHolders. CTM translates EquipmentHolder names. For example, subslot 1 in slot 0 in the NE is named /rack=1/shelf=1/slot=1/sub_slot=2.

For the ONS 15454 MSTP NEs, shelves are numbered using their unique shelf ID, and not their physical location attributes, such as rack number and shelf position. For example, slot 1 on an MSTP shelf with an ID of 7 that is physically located in rack 1, shelf 2 is named /rack=1/shelf=7/slot=1.

For the ONS 155xx family, there is a new equipment holder called a port_holder. Because the TMF determines whether or not a piece of equipment has to be in an EquipmentHolder, the port_holder is created to hold the transceiver modules that contain the PTPs. A port_holder that is present in subslot 1, slot 0 in an NE is named /rack=1/shelf=1/slot=1/sub_slot=2/port_holder=1. A port holder present in slot 1 in an NE is named /rack=1/shelf=1/slot=2/port_holder=1.

For the ONS 15600 NE and ASAP cards, a PIM present in slot 3 is named /rack=1/shelf=1/slot=3/sub_slot=1. A PPM present in PIM 3 in slot 2 is named /rack=1/shelf=1/slot=2/sub_slot=3/ppm_holder=1.

For AdditionalHolder, the naming convention is defined as /additionalHolder=1; there is no subholder for AdditionalHolder. AdditionalHolders apply only to ONS 15454 SONET and SDH NEs when AIE or AEP cards are installed.

For the MGX 8880/8850/8830, there is a new equipment holder called peripheral_holder. Because the TMF determines whether or not a piece of equipment has to be in an EquipmentHolder, the peripheral_holder is created to hold the peripherals.

A peripheral_holder present on a shelf is named rack=1/shelf=1/ /peripheral_holder=17236110, where the number 17236110 is the unique peripheral ID.

A peripheral_holder in slot 1 is named /rack=1/shelf=1/slot=1/peripheral_holder=17236111, where the number 17236111 is the unique peripheral ID.

2.2.5.8 Topological Link

CTM discovers topological links and assigns a name with the following default format:

MEName A:slot number/port number-ME Name B:slot number/port number

Integer values are used for slot and port number. You can use the CTM client to change the link name.

For the ONS 15530 and ONS 15540, the topological link name has the following format, where the port number is optional:

ME NameA:InterfaceType/slot number/subslot number [/port number]-ME NameB:InterfaceType/slot number/subslot number[/port number]

For example:

- NE540A:Wdm0/0--NE540B:Wdm1/0
- NE540A:Wavepatch10/1/0--NE540A:Filter0/3/1

CTM reports two topological links for a Y-cable link. Both topological links have the same nativeEMSName, but different values in the second tuple of the topological link name. The names “<nativeEMSName>:::1” and “<nativeEMSName>:::2” identify the two legs of the Y-cable link.

2.2.5.9 Protection Group

The protection group name reported by the NE is used as the PGP native name for 1_FOR_1, 1_FOR_N, and 1_PLUS_1 protection groups. The ring ID is used as the PGP native name for 2_FIBER_BLSR and 4_FIBER_BLSR protection groups. “<Ring Id>-EAST” and “<Ring Id>-WEST” are used as the native names for the two component groups of 4_FIBER_BLSR PGP.

2.2.5.10 L2 Topology

An L2 topology represents a point-to-point, hub-and-spoke, or Resilient Packet Ring (RPR) Layer 2 network topology. These topologies exist over the underlying Layer 1 topology, which is formed by physical topological links. Each L2 topology name is identified in the context of an EMS and has two tuples. The first tuple represents the EMS name, and the second tuple represents the nativeEMSName of the L2 topology. The L2 topology name is unique within an EMS. Each L2 topology can be associated with multiple ML VLANs.

2.2.5.11 ML VLAN

Each ML VLAN is associated with an L2 topology and is unique for a given L2 topology. The fully qualified ML VLAN name is identified by three tuples. The first tuple represents the EMS name, the second tuple represents the L2 topology name, and the last tuple represents the nativeEMSName of the ML VLAN (which is the VLAN ID). A maximum of 255 ML VLANs can be created per L2 topology, ranging from 1 to 4095.

2.2.5.12 QoS Template

The QoS template is defined in the scope of an EMS domain. The fully qualified QoS template name is identified by two tuples. The first tuple represents the EMS name and the second tuple represents the QoS template name.

2.2.5.13 VCAT

The fully qualified VCAT name is identified by two tuples. The first tuple represents the EMS name and the second tuple represents the VCAT name, which is the same as the nativeEMSName of the VCAT.

2.2.6 Programming Details

2.2.6.1 Boolean Mapping

The following boolean parameters apply:

- On = True
- Off = False

2.2.6.2 Resource Cleanup

Many TMF interfaces that CTM implements return iterator objects for large volumes of data. These iterators are allocated CTM resources. CTM supports a maximum of 128 iterators. If the iterator limit is reached, the NMS receives an EXCPT_TOO_MANY_OPEN_ITERATORS exception.

By default, CTM cleans up all iterator objects if they are not accessed in a time interval of six hours.

2.2.6.3 Development Environment

CTM GateWay/CORBA has been developed with Java Development Kit (JDK) 1.54.2 and jacORB 2.x1.3.7 on Sun Solaris 10 and is compliant with the CORBA 2.3 specification. The OSS can use Java or the C++ IDL compiler to compile IDL files.

Note If you use Java and jacORB, the CTM server installation provides JAR files for notification IDLs and TMF IDLs.

The thread_pool_max property indicates the maximum number of concurrent threads that CTM can handle. By default, the thread_pool_max property is set to 200 in the */CTM-installation-directory*/bin/jcorbagw.sh file. If the value is exceeded, the following exception is returned:

```
org.omg.CORBA.TRANSIENT: Server-side Exception: resource limit reached
```

2.2.6.4 Exception Handling

All interfaces that CTM GateWay/CORBA implements raise the exception globaldefs::ProcessingFailureException. The following sections discuss each exception type in detail.

2.2.6.4.1 EXCPT_NOT_IMPLEMENTED

This exception indicates whether some IDL operations are optional or are not implemented in this release. If the operation itself is not supported, the errorReason is an empty string.

2.2.6.4.2 EXCPT_INTERNAL_ERROR

This exception indicates an internal EMS error and applies to all methods.

2.2.6.4.3 EXCPT_INVALID_INPUT

This exception indicates an incorrect parameter format, such as a three-level naming attribute TP name that is passed as a single-level name. If a parameter is out of range, this exception is also used. The reason field contains the incorrect parameter.

2.2.6.4.4 EXCPT_ENTITY_NOT_FOUND

This exception indicates that the NMS supplied an object name as a parameter and the EMS cannot find an object with that name. The reason field contains the name that was passed as a parameter.

2.2.6.4.5 EXCPT_UNABLE_TO_COMPLY

This exception is used as a generic value when the server cannot respond to the request.

2.2.6.4.6 EXCPT_NE_COMM_LOSS

This exception is used as a generic value when the server cannot communicate with the NE, preventing the successful completion of the operation. All operations that involve communication with the NE might return this exception type.

2.2.6.4.7 EXCPT_ACCESS_DENIED

This exception indicates that an operation has resulted in a security violation. Verify that you have the required access to perform the operation.

2.2.6.4.8 EXCPT_TOO_MANY_OPEN_ITERATORS

This exception indicates that the EMS has exceeded the number of iterators it can support. CTM supports 128 iterators. If you receive this exception, it is possible that other sessions are retrieving large volumes of data. Wait several minutes; then, retry. Alternately, close other sessions. To avoid this exception, the NMS must invoke the destroy method on iterators.

2.2.6.4.9 EXCPT_USERLABEL_IN_USE

This exception indicates that the userLabel uniqueness constraint cannot be met.

2.2.6.4.10 EXCPT_STRICT_MERGE_FAILED

This exception is raised when an OSS issues an upgrade request to merge circuits with a STRICT MERGE operation type. The OSS uses this exception to issue the upgrade request with a LOOSE MERGE operation type.

2.2.7 Provisioning Subnetwork Connections

SNC provisioning must be within a multilayer subnetwork scope, which means the aEnd and the zEnd of an SNC must be in the same multilayer subnetwork.

CTM defines the method multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: createAndActivateSNCFromUserLabel for SNC provisioning.

2.2.7.1 SNC Provisioning on OC-*n* or Electrical Equipment

CTM supports SNC provisioning on OC-*n* or electrical cards for ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, and ONS 15600 SONET NEs.

CTPs are used as the endpoints to create the SNC.

2.2.7.2 SNC Provisioning on Ethernet Equipment

ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, ONS 15454 SONET, and ONS 15454 SDH managed elements support Ethernet equipment. SNCs can be created on this equipment to carry Ethernet traffic. The TMF standard supports only SONET, SDH, DWDM, and ATM; it does not support Ethernet at this time.

Provisioning SNCs on E-series, G-series, and ML-series equipment supported by the ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, ONS 15454 SONET, and ONS 15454 SDH is different from creating SNCs on SONET or PDH equipment. For SNCs on Ethernet equipment, CTM uses FTP to model the endpoint. See 2.2.5.5 Floating Termination Point, page 2-35 for naming conventions.

The three modes supported for SNC provisioning on Ethernet equipment are:

- Single-card mode—E-series cards; VLANs can be associated during creation.
- Multicard mode—E-series cards; VLANs can be associated during creation.
- Linear-card mode—E-series, G-series, and ML-series cards.

To obtain available source and destination FTPs for SNC provisioning, use the following methods:

- managedElementManager::ManagedElementManager_I:: getAllSrcFTPsForSNCProvisioning
- managedElementManager::ManagedElementManager_I:: getAllDestFTPsForSNCProvisioning

2.2.7.2.1 Single-Card Mode and Multicard Mode

By default, all E-series cards are in the multicard mode EtherSwitch group. Each node has only one multicard mode EtherSwitch group; the group exists even if the node has no Ethernet cards. A single-card EtherSwitch group can exist only if a node has at least one Ethernet card.

- To create SNCs in multicard mode, both aEnd and zEnd must be in the multicard EtherSwitch group.
- To create SNCs in single-card mode, one end can be an OC-*n* CTP. You must create a bidirectional SNC.

The following table shows the maximum bandwidth allowed for single- and multicard EtherSwitches.

Table 2-18 Single-Card and Multicard Mode

ONS 15327 Single Card	ONS 15454 SONET/SDH Single Card	ONS 15327 Multicard	ONS 15454 SONET/SDH Multicard
Six STS1s	Six STS1s and two STS3c (SONET only)	Three STS1s	Six STS1s (SONET only)
Two STS3c	Four STS3c (VC4)	One STS3c	Two STS3c (VC4)
One STS6c	One STS6c (VC4_2c) and six STS1s (SONET only)	—	One STS6c (VC4_2c)
One STS12c	One STS6c (VC4_2c) and two STS3c (VC4)	—	—
—	Two STS6c (VC4_2c)	—	—
—	One STS12c (VC4_4c)	—	—

2.2.7.2.2 Linear-Card Mode

You can apply linear-card mode SNC provisioning to all types of Ethernet cards. One end of the SNC must be an FTP; the other end can be a CTP on any OC-*n* port, or an FTP. The FTPs must use linear-mode naming conventions.

Only single source and single destination are supported for the SNC creation, and the SNC must be bidirectional.

For E-series and G-series cards, the port number in the FTP name is the number of the front Ethernet port. For ML-series cards, the port number is the number of the virtual back-end port.

2.2.7.2.3 Layer Rates

CTM supports the following layer rates on E-series cards:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4
- LR_STS6c_and_VC4_2c
- LR_STS12c_and_VC4_4c

CTM supports the following layer rates on G-series cards:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4
- LR_STS6c_and_VC4_2c
- LR_STS9c_and_VC4_3c
- LR_STS12c_and_VC4_4c
- LR_STS24c_and_VC4_8c
- LR_STS48c_and_VC4_16c

CTM supports the following layer rates on ML-series cards:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4
- LR_STS6c_and_VC4_2c
- LR_STS9c_and_VC4_3c
- LR_STS12c_and_VC4_4c
- LR_STS24c_and_VC4_8c

CTM supports the following layer rates on FCMR cards:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4

CTM supports the following layer rate on CE-100T-8 and ML-100T-8 cards:

- LR_STS1_and_AU3_High_Order_VC3

2.2.7.2.4 SNC Provisioning for ONS 15454 CE-MR-10 Cards

The ONS 15454 CE-MR-10 card implements two different working modes:

- Automatic mode—Available only for ONS 15454 SONET
- Manual (or Exposed) mode—Available for ONS 15454 SONET and ONS 15454 SDH

Automatic mode is similar to the provisioning of existing ML-series cards, where FTPs must be used for CCAT and VCAT circuit provisioning, and the FTP name is the number of the virtual back-end port.

Manual mode requires PTPs/CTPs to be specified for the aEnd and zEnd on CCAT circuits, and for aEnd(s) and zEnd(s) on VCAT member circuits. For VCAT circuit endpoints, FTPs must be used.

Use one of the following APIs to retrieve available CTPs:

- 3.8.20 getAllSrcCTPsForVCATMemberSNCProvisioning, page 3-104
- 3.8.23 getAllDestCTPsForVCATMemberSNCProvisioning, page 3-107

Use the API getAllSrc*PTP/CTP/FTP*ForCircuitProvisioning to retrieve the possible endpoints, taking into account the current card mode.

Note You cannot change the card mode from the CTM GateWay/CORBA interface. You cannot change the card mode if there is a circuit on the card.

To determine whether a CE-MR-10 card is configured for Manual or Automatic mode, use the getEquipment API, which returns the CardMode parameter.

The following layer rates are supported for CCAT circuits:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4
- LR_STS6c_and_VC4_2c
- LR_STS9c_and_VC4_3c
- LR_STS12c_and_VC4_4c
- LR_STS24c_and_VC4_8c
- LR_STS48c_and_VC4_16c

The following layer rates are supported for VCAT member circuits:

- LR_STS1_and_AU3_High_Order_VC3
- LR_STS3c_and_AU4_VC4
- LR_VT1_5_and_TU11_VC11

2.2.7.2.5 SNC Provisioning for ONS 15454 ADM-10G Cards

The ADM_10G card ports carry the OCN and GE traffic. You can provision each of the first eight ports as 1GE port that carry GE traffic. These ports, by default, GFP-encapsulation enabled. However, if required, you can configure them as HDLC.

CTM Gateway/CORBA can manage CCAT bidirectional SNCs with FTP termination points on ONE_GE_PORT ports ranging from 1 to 8 in the framing protocol types previously listed.

The following table lists the supported layer rates along with their corresponding values:

Table 2-19 Layer Rates Supported by ONS 15454 ADM-10G Cards

Layer Rates	Values
STS1	LR_STS1_and_AU3_High_Order_VC3
STS3C	LR_STS3c_and_AU4_VC4
STS6C	LR_STS6c_and_VC4_2c
STS9C	LR_STS9c_and_VC4_3c
STS12C	LR_STS12c_and_VC4_4c
STS24C	LR_STS24c_and_VC4_8c

As opposed to the MSPP data cards and the OC cards, FTPs on ADM_10G client ports contain reference to ppm_holder as well. For example:

```
15454-ANSI-68-8
/rack=1/shelf=1/slot=5/ppm_holder=2/port=1
```

2.2.7.3 Tunnel SNC Provisioning

CTM supports tunnel SNC provisioning on ONS 15310, ONS 15327, ONS 15454 SONET, and ONS 15454 SDH NEs. The tunnel SNC must be bidirectional.

The tunnel SNC for the ONS 15310, ONS 15327, and ONS 15454 SONET is called VT tunnel, and the layer rate must be 14 (LR_STS1_and_AU3_High_Order_VC3). The tunnel SNC for the ONS 15454 SDH is called VC tunnel, and the layer rate must be 15 (LR_STS3c_and_AU4_VC4). See 2.2.5 Naming Conventions, page 2-30 for naming conventions.

Only one aEnd and zEnd are specified for the VT/VC tunnel SNC provisioning.

CTM supports only fully automatically routed VT/VC tunnel SNCs. CTM does not support manually routed and constraint-based automatically routed VT/VC tunnel SNCs.

CTM supports VC tunnel for VC3 port grouping and SNC provisioning on ONS 15454 SDH. For VC tunnel for VC3 port grouping and SNC provisioning, if the source or destination end is an STM-*n* card, it is modeled as CTP; if the source or destination end is on an E3, DS3I, or DS3IN card, it is modeled as FTP.

The following table shows the combinations of cards that can be selected for creating the VC4 tunnel for VC3 port grouping circuits.

Table 2-20 Tunnel SNC Provisioning

A-End	Z-End
DS3I or DS3IN	DS3I, DS3IN, or STM-*n*
E3	E3 or STM-*n*
STM-*n*	DS3I, DS3IN, or E3

When VC tunnel for VC3 port grouping and SNC provisioning is complete:

- One VC4 tunnel SNC is created with the SNC name *name*:PGT1
- Three VC3 SNCs are created with the SNC names *name*:PGC1, *name*:PGC2, and *name*:PGC3

Deleting one PGC SNC results in the deletion of all four SNCs. You cannot delete a PGT SNC if any PGC SNCs exist. See 2.2.5 Naming Conventions, page 2-30 for the naming conventions.

2.2.7.4 SNC Provisioning on WDM Equipment

2.2.7.4.1 OCHCC Circuit Provisioning

CTM supports OCHCC circuit provisioning on transponders, muxponders, and ITU-T line cards for ONS 15454 SONET and ONS 15454 SDH NEs.

The OCHCC SNC is created between client ports on transponder or muxponder cards, or on trunk ports on ITU-T line cards.

OCHCC is an extension of OCHNC. OCHCC is a part of the OCHCC SNC. OCHCC can have only one source and one destination.

CTP naming is identical to OCHNC SNCs: CTM GateWay/CORBA determines whether to create an OCHNC or OCHCC, depending on the layer. See 2.2.5.4 Connection Termination Point, page 2-31 for the CTP details.

OCHCC must be bidirectional.

CTM supports different sizes corresponding to possible payload types of transponder and muxponder client ports. A specific layer must be defined for each payload type.

OCHCC is either *unprotected* or *protected*. Protected is created only on protected transponder or muxponder cards where there is a protected trunk port.

Note To create OCHCC, you must create patchcord links at an optical channel layer between the transponders, muxponders, or ITU-T line cards and DWDM ports.

OSPF-detected links between GCC terminations on transponder or muxponder cards are indicated at the Optical_Transport_OTUk layer.

When an OCHCC circuit is created, unless transponder cards are directly connected by a trunk-to-trunk patchcord link, an additional OCH trail tunnel circuit is created automatically.

The following method is defined for protected OCHCC circuits when it is necessary to specify constraints for both working and protected routes:
multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::
createAndActivateProtectedOchccFromUserLabel

2.2.7.4.2 OCHNC Circuit Provisioning

CTM supports OCHNC circuit provisioning on the following WDM cards for ONS 15454 SONET and ONS 15454 SDH NEs:

- 40-SMR1-C
- 40-SMR2-C
- AD_1B_CARD_EQPT
- AD_1C_CARD_EQPT
- AD_2C_CARD_EQPT
- AD_4B_CARD_EQPT
- AD_4C_CARD_EQPT
- AMP_PORT_EQPT
- DMX32_L_CARD_EQPT
- DMX32_O_CARD_EQPT
- DMX32_R_CARD_EQPT
- DMX40_LINE_CARD_EQPT
- MMU_CARD_EQPT
- MUX_DMX_PORT_EQPT
- MUX_DMX4_CARD_EQPT
- MUX32_O_CARD_EQPT
- MUX40_LINE_CARD_EQPT

- OADM_PORT_EQPT
- OPT_AMP_17_C_LINE_CARD_EQPT
- OPT_AMP_L_EQPT
- OPT_BST_E_EQPT
- OPT_BST_EQPT
- OPT_BST_L_EQPT
- OPT_PRE_EQPT
- OSC_CSM_CARD_EQPT
- OSCM_CARD_EQPT
- OSCM_PORT_EQPT
- WSS_32_CARD_EQPT
- WSS32_L_CARD_EQPT
- WSS40_LINE_CARD_EQPT
- WXC40_LINE_CARD_EQPT

This type of circuit is referred to as WDM SNC. The WDM SNC is created on a particular wavelength channel with a direction of either east-to-west or west-to-east. WDM SNCs are unidirectional or bidirectional. CTM GateWay/CORBA supports the creation of bidirectional OCHNC circuits on R5.0 and later NEs. You can also create two unidirectional OCHNC circuits on the same frequency, in opposite directions.

The CTPs used in WDM SNCs are different from the CTPs used in SONET/SDH SNCs in terms of naming and nature. See 2.2.5.4 Connection Termination Point, page 2-31 for the CTP naming conventions. The layer rate for the CTPs and WDM SNC must be LR_Optical_Channel.

The OCHNC size is “Equipped not specific.” You can specify four possible directions:

- East-to-west unidirectional
- East-to-west bidirectional
- West-to-east unidirectional
- West-to-east bidirectional

The preceding values are added to the globaldefs::ConnectionDirection_T structure.

For OCHNC on ONS 15454 SONET and ONS 15454 SDH R7.0 and later, you do not have to specify the east-to-west or west-to-east direction; instead, UniDir and BiDir must be used. For R8.0 NEs you can specify route constraints (NE include/exclude constraints) in the SNCCreateData_T parameter.

When creating OCHNC bidirectional circuits on R5.x or R6.x NEs, you must specify all four circuit endpoints: source, destination, secondary source (the source in the opposite direction), and secondary destination (the destination in the opposite direction). For R7.x or R8.x NEs you must specify only the source and destination in one direction, because the endpoints for the opposite direction are calculated automatically.

However, with the 80-WXC card (introduced in R9.2), more secondary sources and secondary destinations could be available. In such a case, you must also specify the secondary TP. If they do not belong to the actual list of available PTPs, an EXCPT_UNABLE_TO_COMPLY exception is thrown. For example:

```
The secondary destination does not belong to the list of available PTPs:
/rack=1/shelf=1/slot=4/port=1, /rack=1/shelf=1/slot=4/port=2,
/rack=1/shelf=1/slot=4/port=3, /rack=1/shelf=1/slot=4/port=4,
```

```
/rack=1/shelf=1/slot=4/port=5, /rack=1/shelf=1/slot=4/port=6,
/rack=1/shelf=1/slot=4/port=7, /rack=1/shelf=1/slot=4/port=8,
/rack=1/shelf=1/slot=4/port=9
```

Or:

```
The secondary source does not belong to the list of available PTPs:
/rack=1/shelf=1/slot=2/port=9
```

A mixed configuration, such as an OCHNC bidirectional circuit between an 80WXC card (configured as bidirectional) and a MUX/DMX card, requires only the secondary drop on the 80WXC card. CTM still automatically calculates the secondary source on the MUX/DMX card.

For the ONS 15454 SONET and ONS 15454 SDH R9.0 and later, OCHNC circuits support protected optical paths. You can specify the following property within the creation data structure: staticProtectionLevel=StaticProtectionLevel_T.FULLY_PROTECTED.

The availability of protected paths and the possibility of provisioning protected OCHNC circuits depend on the presence of the Protection Switch Module (PSM). When configured correctly, the PSM provides one of the available protection schemas.

If the PSM card mode is configured as STANDALONE, you cannot create OCHNC or OCHCC protected circuits that involve the PSM card. The STANDALONE mode can be set and retrieved from the CTM client GUI only. (In the CTM GUI, launch the NE Explorer, select the PSM card, click the Card tab, and configure the Card Mode attribute.)

CTM GateWay/CORBA distinguishes the mode that the NE version uses.

2.2.7.5 Specifying Route Constraints for OCH*xx* Circuit Provisioning

For OCHCC, OCHNC, and OCH trail tunnel circuits, you can specify route constraints (NE include/exclude constraints) in the SNCCreateData_T parameter.

For protected OCHCC circuits, constraints for the working path are specified in the SNCCreateData_T parameter; constraints for the protected path are specified in a separate parameter. See details in the createAndActivateProtectedOchccFromUserLabel API description.

2.2.7.6 Low-Order Circuit Provisioning Using VAP/LAP

CTM supports low-order circuit provisioning using VAP circuits on ONS 15310 CL, ONS 15310 MA SONET, ONS 15310 MA SDH, ONS 15327, and ONS 15454 SONET NEs, and low-order aggregation point (LAP) circuits on ONS 15454 SDH NEs. The VAP/LAP circuits must be bidirectional.

When provisioning VAP/LAP circuits, the source is treated as the STS grooming end; the destination is treated as the VT grooming end and is used later to connect VT circuits.

The source is a CTP and the destination is an FTP. See 2.2.5 Naming Conventions, page 2-30 for naming conventions.

CHAPTER 3

Using CTM GateWay/CORBA Interfaces

This chapter describes each interface that CTM supports, including TeleManagement Forum (TMF)-specific and Cisco-proprietary interfaces. This chapter includes the following information:

- 3.1 Overview of APIs Supported by NE Type, page 3-2
- 3.2 common::Common_I, page 3-18
- 3.3 emsMgr::EMSMgr_I, page 3-19
- 3.4 EmsSession::EmsSession_I, page 3-50
- 3.5 emsSessionFactory::EmsSessionFactory_I, page 3-52
- 3.6 equipment::EquipmentInventoryMgr_I, page 3-55
- 3.7 maintenanceOps::MaintenanceMgr_I, page 3-79
- 3.8 managedElementManager::ManagedElementManager_I, page 3-82
- 3.9 multiLayerSubnetwork::MultiLayerSubnetworkMgr_I, page 3-123
- 3.10 L2 Topology and ML VLAN for ML-Series Ethernet Cards, page 3-180
- 3.11 E-Series VLAN Interfaces, page 3-205
- 3.12 nmsSession::NmsSession_I, page 3-208
- 3.13 performance::PerformanceManagementMgr_I, page 3-210
- 3.14 protection::ProtectionMgr_I, page 3-276
- 3.15 session::Session_I, page 3-287
- 3.16 mtnmVersion::Version_I, page 3-289
- 3.17 Transmission Descriptor Interfaces, page 3-289
- 3.18 Software and Data Manager Interface, page 3-295

Note The iterator interfaces defined by the TMF are not listed here. CTM implementation of these interfaces does not deviate from the TMF definition.

3.1 Overview of APIs Supported by NE Type

The following table lists each API and the NEs that support it.

Table 3-1 APIs Supported by NE Type

API No.	CTM GateWay/CORBA API	Supported NEs
1.	common::Common_I:: getCapabilities	—
2.	emsMgr::EMSMgr_I:: getAllEMSAndMEActiveAlarms	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
3.	emsMgr::EMSMgr_I:: getAllEMSSystemActiveAlarms	—
4.	emsMgr::EMSMgr_I:: getAllTopLevelSubnetworks	—
5.	emsMgr::EMSMgr_I:: getAllTopLevelSubnetworkNames	—
6.	emsMgr::EMSMgr_I:: getAllTopLevelTopologicalLinks	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
7.	emsMgr::EMSMgr_I:: getAllTopLevelTopologicalLinkNames	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
8.	emsMgr::EMSMgr_I:: getTopLevelTopologicalLink	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
9.	emsMgr::EMSMgr_I:: createTopologicalLink	MDS 9000, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15600 SDH, ONS 15600 SONET, unmanaged NE
10.	emsMgr::EMSMgr_I:: createTopologicalLinkWithYCableProtection	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
11.	emsMgr::EMSMgr_I:: deleteTopologicalLink	MDS 9000, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET, unmanaged NE
12.	emsMgr::EMSMgr_I:: modifyTopologicalLink	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
13.	emsMgr::EMSMgr_I:: getEMS	—
14.	emsMgr::EMSMgr_I:: getAllL2Topologies	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
15.	emsMgr::EMSMgr_I:: getAllQoSTemplateNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
16.	emsMgr::EMSMgr_I:: getAllQoSTemplates	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
17.	emsMgr::EMSMgr_I:: getQoSTemplate	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
18.	emsMgr::EMSMgr_I:: getAllEMSandMEClearedAlarms	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
19.	emsMgr::EMSMgr_I:: getAllEMSSystemClearedAlarms	All
20.	emsMgr::EMSMgr_I:: createServerTrailLink	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15600 SDH, ONS 15600 SONET
21.	emsMgr::EMSMgr_I:: deleteServerTrailLink	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15600 SDH, ONS 15600 SONET
22.	emsMgr::EMSMgr_I:: addNetworkPartition	—
23.	emsMgr::EMSMgr_I:: addMultiLayerSubnetwork	All
24.	emsMgr::EMSMgr_I:: deleteNetworkPartition	—
25.	emsMgr::EMSMgr_I:: deleteMultiLayerSubnetwork	All
26.	emsMgr::EMSMgr_I:: getAllSupportedMEModels	All
27.	emsSession::EmsSession_I:: getEventChannel	—
28.	emsSession::EmsSession_I:: getManager	All

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
29.	emsSession::EmsSession_I:: getSupportedManagers	All
30.	emsSessionFactory::EmsSessionFactory_I:: getEmsSession	—
31.	emsSessionFactory::EmsSessionFactory_I:: getEmsPublicKey	—
32.	emsSessionFactory::EmsSessionFactory_I:: getEmsPublicKeyPair	—
33.	equipment::EquipmentInventoryMgr_I:: getAllEquipment	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
34.	equipment::EquipmentInventoryMgr_I:: getAllEquipmentNames	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
35.	equipment::EquipmentInventoryMgr_I:: getAllSupportedPTPs	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
36.	equipment::EquipmentInventoryMgr_I:: getAllSupportedPTPNames	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
37.	equipment::EquipmentInventoryMgr_I:: getAllSupportingEquipment	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
38.	equipment::EquipmentInventoryMgr_I:: getAllSupportingEquipmentNames	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
39.	equipment::EquipmentInventoryMgr_I:: getContainedEquipment	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2, ONS 15454-M6, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
40.	equipment::EquipmentInventoryMgr_I:: getEquipment	MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
41.	equipment::EquipmentInventoryMgr_I:: provisionEquipment	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
42.	equipment::EquipmentInventoryMgr_I:: setAlarmReportingOff	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15600 SDH, ONS 15600 SONET
43.	equipment::EquipmentInventoryMgr_I:: setAlarmReportingOn	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
44.	equipment::EquipmentInventoryMgr_I:: unprovisionEquipment	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
45.	maintenanceOps::MaintenanceMgr_I:: getActiveMaintenanceOperations	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SONET
46.	maintenanceOps::MaintenanceMgr_I:: performMaintenanceOperation	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
47.	managedElementManager::ManagedElementManager_I::getAllActiveAlarms	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
48.	managedElementManager::ManagedElementManager_I::getAllSNCs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
49.	managedElementManager::ManagedElementManager_I::getAllManagedElements	All
50.	managedElementManager::ManagedElementManager_I::getAllManagedElementNames	All
51.	managedElementManager::ManagedElementManager_I::getAllPTPs	MDS 9000, MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
52.	managedElementManager::ManagedElementManager_I::getAllPTPNames	MDS 9000, MGX 8880/8850/8830, ONS 15216, ONS 15305, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
53.	managedElementManager::ManagedElementManager_I::getContainedInUseTPs	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
54.	managedElementManager::ManagedElementManager_I::getContainedInUseTPNames	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
55.	managedElementManager::ManagedElementManager_I::getContainedPotentialTPs	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
56.	managedElementManager::ManagedElementManager_I::getContainedPotentialTPNames	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
57.	managedElementManager::ManagedElementManager_I::getContainingSubnetworkNames	All
58.	managedElementManager::ManagedElementManager_I::getManagedElement	All
59.	managedElementManager::ManagedElementManager_I::getTP	ONS 15216, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
60.	managedElementManager::ManagedElementManager_I::getAllCtmMEInternalPatchord	ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
61.	managedElementManager::ManagedElementManager_I::setAdditionalInfo	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
62.	managedElementManager::ManagedElementManager_I::getAssociatedTopologicalLinks	ONS 15216, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
63.	managedElementManager::ManagedElementManager_I::setTPData	ONS 15216, ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
64.	managedElementManager::ManagedElementManager_I::getAllSrcPTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
65.	managedElementManager::ManagedElementManager_I::getAllSrcCTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
66.	managedElementManager::ManagedElementManager_I::getAllDestPTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, OONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
67.	managedElementManager::ManagedElementManager_I::getAllDestCTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
68.	managedElementManager::ManagedElementManager_I::getAllSrcFTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
69.	managedElementManager::ManagedElementManager_I::getAllDestFTPsForSNCProvisioning	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
70.	managedElementManager::ManagedElementManager_I::forceNEResync	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
71.	managedElementManager::ManagedElementManager_I::getAllTopologicalLinksForME	ONS 15216, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
72.	managedElementManager::ManagedElementManager_I::getAllRolls	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
73.	managedElementManager::ManagedElementManager_I::getAllRollNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
74.	managedElementManager::ManagedElementManager_I::getRoll	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
75.	managedElementManager::ManagedElementManager_I::getRollToPTPs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
76.	managedElementManager::ManagedElementManager_I::getRollToCTPs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
77.	managedElementManager::ManagedElementManager_I::getRollFromCTPs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
78.	managedElementManager::ManagedElementManager_I::setPortName	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
79.	managedElementManager::ManagedElementManager_I::getPortName	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
80.	managedElementManager::ManagedElementManager_I::getAllClearedAlarms	ONS 15305, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
81.	managedElementManager::ManagedElementManager_I::getAllCrossConnections	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SONET
82.	managedElementManager::getAllSoftwareDownloadJobs	—
83.	managedElementManager::ManagedElementMgr_I::setMEAdminState	All
84.	managedElementManager::ManagedElementManager_I::getAllSrcCTPsForVCATMemberSNCProvisioning	ONS 15454 SDH, ONS 15454 SONET
85.	managedElementManager::ManagedElementManager_I::getAllDestCTPsForVCATMemberSNCProvisioning	ONS 15454 SDH, ONS 15454 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
86.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllEdgePoints	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
87.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllEdgePointNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
88.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllManagedElement	All
89.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllManagedElementNames	All
90.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllSubnetworkConnections	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
91.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllSubnetworkConnectionsWithTP	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
92.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllSubnetworkConnectionNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
93.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllSubnetworkConnectionNamesWithTP	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
94.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllTopologicalLinks	ONS 15216, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
95.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllTopologicalLinkNames	ONS 15216, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
96.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getMultiLayerSubnetwork	All
97.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getCTPsAndTopologicalLinks	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
98.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getSNC	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
99.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getSNCsByUserLabel	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
100.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getTopologicalLink	ONS 15216, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15530, ONS 15540, ONS 15600 SDH, ONS 15600 SONET
101.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createAndActivateSNCFromUserLabel	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
102.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createAndActivateSNC	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
103.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createAndActivateProtectedOchccFromUserLabel	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
104.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::addDrops	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
105.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deactivateAndDeleteSNC	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
106.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getVCATSNC	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
107.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllVCATSNCs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
108.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllVCATSNCNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
109.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllVCATMemberSNCs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
110.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllVCATMemberSNCNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
111.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createVCAT	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
112.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteVCAT	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
113.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::addMemberSNCsToVCAT	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
114.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteMemberSNCsFromVCAT	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
115.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteAllMemberSNCFromVCAT	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
116.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getValidSubnetworkConnectionsForMerge	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
117.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::upgrade	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
118.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createRoll	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
119.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteRoll	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
120.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::cancelRoll	Release 6.0 of Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
121.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::forceValidSignal	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
122.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::completeRoll	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
123.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::finishRoll	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
124.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllRolls	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
125.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllRollNames	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
126.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAssociatedRollNamesForSNC	Release 6.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
127.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAssociatedSNCNamesForRoll	Release 5.0 of ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
128.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: createAndActivateSNCOnDRI	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
129.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: addManagedElement	All
130.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: deleteManagedElement	All
131.	nmsSession::NmsSession_I:: historyPMDataCompleted	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
132.	nmsSession::NmsSession_I:: historyPMDataFailed	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
133.	nmsSession::NmsSession_I:: alarmLossOccurred	—
134.	performance::PerformanceManagementMgr_I:: disablePMData	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
135.	performance::PerformanceManagementMgr_I:: enablePMData	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
136.	performance::PerformanceManagementMgr_I:: setTCATPParameter	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
137.	performance::PerformanceManagementMgr_I:: getTCATPParameter	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
138.	performance::PerformanceManagementMgr_I:: getCTMHistoryPMData	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
139.	performance::PerformanceManagementMgr_I:: deleteHistoryPMDataFile	ONS 15310 CL, ONS 15327, ONS 15454 SDH, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SONET
140.	performance::PerformanceManagementMgr_I:: getTPHistoryPMData	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
141.	performance::PerformanceManagementMgr_I:: getHoldingTime	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
142.	performance::PerformanceManagementMgr_I:: getAllCurrentPMData	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
143.	performance::PerformanceManagementMgr_I:: getMEPMcapabilities	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
144.	protection::ProtectionMgr_I:: getAllProtectedTPNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
145.	protection::ProtectionMgr_I:: getAllProtectionGroups	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
146.	protection::ProtectionMgr_I:: getProtectionGroup	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
147.	protection::ProtectionMgr_I:: performProtectionCommand	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327,ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
148.	protection::ProtectionMgr_I:: createProtectionGroup	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
149.	protection::ProtectionMgr_I:: deleteProtectionGroup	ONS 15305 CTC, ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
150.	protection::ProtectionMgr_I:: getAssociatedProtectionGroup	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET, ONS 15600 SDH, ONS 15600 SONET
151.	session::Session_I:: endSession	—
152.	session::Session_I:: ping	—
153.	mtnmVersion::Version_I:: getVersion	—
154.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: createMLVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
155.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANAddFrontPorts	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
156.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANDropFrontPorts	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
157.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANPortState	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
158.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANRSTPState	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
159.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANTPQOS	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
160.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: modifyMLVLANCustomerIDServiceID	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
161.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: getMLVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
162.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: deleteMLVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
163.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: getAllMLVLANs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
164.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: getAllMLVLANNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
165.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllAssociatedSNCs	ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET,
166.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getL2Topology	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
167.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAssociatedL2TopologyForSNC	ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
168.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAssociatedL2TopologyForMLVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
169.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createL2Topology	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
170.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteL2Topology	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
171.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::modifyL2Topology	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
172.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllAvailableMLEquipmentOrTPNames	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454 SONET
173.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::createVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
174.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::deleteVLAN	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
175.	multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getAllVLANs	ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15454 SDH, ONS 15454-M2 SDH, ONS 15454-M6 SDH, ONS 15454 SONET, ONS 15454-M2 SONET, ONS 15454-M6 SONET
176.	softwareAndDataManager::SoftwareAndDataMgr_I::backupME	All
177.	softwareAndDataManager::SoftwareAndDataMgr_I::getMEBackupStatus	All
178.	softwareAndDataManager::SoftwareAndDataMgr_I::abortMEBackup	All
179.	softwareAndDataManager::SoftwareAndDataMgr_I::getBackupList	All
180.	softwareAndDataManager::SoftwareAndDataMgr_I::activateSoftwareOnME	All
181.	softwareAndDataManager::SoftwareAndDataMgr_I::revertSoftwareOnME	All

Table 3-1 APIs Supported by NE Type (continued)

API No.	CTM GateWay/CORBA API	Supported NEs
182.	softwareAndDataManager::SoftwareAndDataMgr_I:: restoreME	All
183.	softwareAndDataManager::SoftwareAndDataMgr_I:: getMERestoreStatus	All
184.	softwareAndDataManager::SoftwareAndDataMgr_I:: abortMERestore	All
185.	softwareAndDataManager::SoftwareAndDataMgr_I:: getAllSupportedMESoftwareVersions	All
186.	softwareAndDataManager::SoftwareAndDataMgr_I:: downloadMESoftware	All
187.	softwareAndDataManager::SoftwareAndDataMgr_I:: addSupportedMESoftwareVersion	All
188.	softwareAndDataManager::SoftwareAndDataMgr_I:: getSWDownloadStatus	All
189.	performance::PerformanceManagementMgr_I:: getCTMValidHistoryPMData	All

3.2 common::Common_I

The Common_I interface is a set of services and utilities that each manager interface inherits. CTM supports only one method from this interface.

3.2.1 common::Common_I:: getCapabilities

Synopsis

```
void getCapabilities(out CapabilityList_T capabilities)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the capabilities of the manager. All nonspecified capabilities are assumed to be unsupported.

Parameters

Name	Type	Input/Output	Description
capabilities	CapabilityList_T	Out	List of capabilities of this manager object. It is a name-value pair, where the name represents the feature or capability and the value represents the support or nonsupport of the specified feature or capability.

A Capability_T value is used to identify a functionality supported by the EMS across the network management layer-element management layer (NML-EML) interface. It is a name/value pair, in which the name represents the feature/capability name and the value represents the support or nonsupport of the specified feature/capability.

The EMS capabilities for this release include individual IDL operation support. The feature/capability name part is used to identify an IDL operation using the following convention:

module_name::interface_name::operation_name

The currently defined values are:

- Supported—The specified feature/capability is fully or partially supported across the NML-EML interface; an operation may be partially supported if not all values of the parameters are supported.
- Unsupported—The specified feature/capability is not supported across the NML-EML interface.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

TMF-defined.

3.3 emsMgr::EMSMgr_I

The EMSMgr_I interface accesses operations that deal with the EMS itself. A handle to an instance of this interface is gained through the emsSession::EmsSession_I::getManager operation in emsSession. This interface is inherited from common::Common_I.

```
interface EMSMgr_I:common::Common_I
```

This interface implements the getCapabilities method. Operations described in the following sections are returned to the NMS when it invokes getCapabilities.

3.3.1 getAllEMSAndMEActiveAlarms

Synopsis

```
void getAllEMSAndMEActiveAlarms(
        in notifications::ProbableCauseList_T excludeProbCauseList,
        in notifications::PerceivedSeverityList_T excludeSeverityList,
        in unsigned long how_many,
        out notifications::EventList_T eventList,
        out notifications::EventIterator_I eventIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request active alarms that are under the control of the EMS (alarms raised by the NEs and alarms raised by the EMS itself). Some alarms might be filtered out (excluded) by specifying their probable causes or severities.

Due to performance concerns, only one OSS client is allowed to invoke this operation at a time. Otherwise, CTM throws the EXCPT_UNABLE_TO_COMPLY exception.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

ONS 15310, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15600 SONET, and ONS 15600 SDH

ONS 15310, ONS 15327, ONS 15454 SDH, ONS 15454 SONET, ONS 15600 SONET, and ONS 15600 SDH do not send a “clear” for TCA events. TCAs are not reported in the eventList returned by this method.

ONS 15540 and ONS 15530

TCAs are reported in the eventList returned by this method. The ONS 15540 sends a “clear” for TCAs.

Parameters

Name	Type	Input/Output	Description
excludeProbCauseList	notifications::ProbableCauseList_T	In	List of probable causes to exclude (for which events are not reported).
excludeSeverityList	notifications::PerceivedSeverityList_T	In	List of severities to exclude from the output event list.
how_many	unsigned long	In	Maximum number of alarms to report in the first batch.
eventList	notifications::EventList_T	Out	First batch of alarms.
eventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_UNABLE_TO_COMPLY - Raised when this method is being used by another OSS client.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.2 getAllEMSSystemActiveAlarms

Synopsis

```
void getAllEMSSystemActiveAlarms(
        in notifications::PerceivedSeverityList_T excludeSeverityList,
        in unsigned long how_many,
        out notifications::EventList_T eventList,
        out notifications::EventIterator_I eventIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all active alarms for the EMS. Some alarms can be filtered out (excluded) by specifying their severities.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

Parameters

Name	Type	Input/Output	Description
excludeSeverityList	notifications::PerceivedSeverityList_T	In	List of severities to exclude from the output event list.
how_many	unsigned long	In	Maximum number of alarms to report in the first batch.
eventList	notifications::EventList_T	Out	First batch of alarms.
eventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.3 getAllTopLevelSubnetworks

Synopsis

```
void getAllTopLevelSubnetworks(
         in unsigned long how_many,
         out multiLayerSubnetwork::SubnetworkList_T sList,
         out multiLayerSubnetwork::SubnetworkIterator_I sIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all subnetworks that are under the control of the EMS.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of subnetworks to return in the first batch.
sList	multiLayerSubnetwork::SubnetworkList_T	Out	First batch of subnetworks.
sIt	multiLayerSubnetwork::SubnetworkIterator_I	Out	Iterator to retrieve the remaining subnetworks.

CTM support for MultiLayerSubnetwork_T is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field contains two tuples. The first tuple identifies the EMS. The second tuple is the same as nativeEMSName. The naming convention is described in 2.2.5.1 Multilayer Subnetwork, page 2-30.
- string userLabel
  - Supported.

- string nativeEMSName
  - Supported.
- string owner
  - Not supported.
- Topology_T subnetworkType
  - Supports TOPO_SINGLETON, TOPO_CHAIN, TOPO_PSR, TOPO_OPEN_PSR, TOPO_SPRING, TOPO_OPEN_SPRING, and TOPO_MESH.
- transmissionParameters::LayerRateList_T supportedRates
  - Not supported.
- globaldefs::NVSList_T additionalInfo
  - Supported.

For the CTM client subnetwork types, the following mapping applies:

CORBA Subnetwork Type	CTM Client Subnetwork Type
TOPO_SINGLETON	SINGLETON
TOPO_CHAIN	LINEAR
TOPO_PSR	UPSR
TOPO_PSR	SNC
TOPO_OPEN_PSR	OPEN_UPSR
TOPO_OPEN_PSR	OPEN_SNC
TOPO_SPRING	BLSR
TOPO_SPRING	MSSP
TOPO_OPEN_SPRING	OTHER
TOPO_MESH	MESH
TOPO_MESH	UNKNOWN
TOPO_MESH	DEFAULT

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.4 getAllTopLevelSubnetworkNames

Synopsis

```
void getAllTopLevelSubnetworkNames(
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the interface described in 3.3.3 getAllTopLevelSubnetworks, page 3-21, but instead of returning entire object structures, this interface returns their names.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of subnetworks to return in the first batch.
sList	globaldefs::NamingAttributesList_T	Out	First batch of subnetwork names.
sIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining subnetwork names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.5 getAllTopLevelTopologicalLinks

Synopsis

```
void getAllTopLevelTopologicalLinks(
         in unsigned long how_many,
         out topologicalLink::TopologicalLinkList_T topoList,
         out topologicalLink::TopologicalLinkIterator_I topoIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the top-level topological links that are under the control of the EMS. A top-level topological link is a link in which the link endpoints exist in two different multilayer subnetworks.

If a topological link that is under the control of the EMS is removed from the NE, the provType parameter is reported as INVALID.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of top-level topological links to return in the first batch.
topoList	topologicalLink::TopologicalLinkList_T	Out	First batch of top-level topological links.
topoIt	topologicalLink::TopologicalLinkIterator_I	Out	Iterator to retrieve the remaining top-level topological links.

CTM support for TopologicalLink_T structure is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field has two tuples. The first tuple is for the EMS. The second is for the topological link. The value in the second tuple is the same as the nativeEMSName field except for the Y-cable links, which have "::1" and "::2" appended to the nativeEMSName to identify the two legs of the Y-cable link.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported. For more information, see 2.2.5 Naming Conventions, page 2-30.
- string owner
  - Not supported.
- globaldefs::ConnectionDirection_T direction. CTM supports CD_BI for bidirectional links and CD_UNI for unidirectional links.
- transmissionParameters::LayerRate_T rate. Supported. For a list of possible values, see 2.2.1 Layer Rate, page 2-23.
- globaldefs::NamingAttributes_T aEndTP
  - Supported.
- globaldefs::NamingAttributes_T zEndTP
  - Supported.
- globaldefs::NVSList_T additionalInfo
  - Supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support topological links	provType	AUTODISCOVERED MANUAL UNMANAGED PATCHCORD SERVER_TRAIL
All managed elements that support topological links	linkStatus	LINK_VALID LINK_INVALID

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support topological links	linkProtectionType	PROT_ONE_PLUS_ONE PROT_BLSR_2F PROT_BLSR_4F PROT_BLSR_2F_PCA PROT_BLSR_4F_PCA PROT_PCA PROT_TUNNEL PROT_UNPROTECTED PROT_UNKNOWN PROT_NOT_APPLICABLE PROT_PARTIALLY_PROTECTED PROT_YCABLE
All managed elements that support topological links	linkCost	Integer, 0 to 1024
ONS 15530 and ONS 15540	connectionType	Inter-NE, Intra-NE.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.6 getAllTopLevelTopologicalLinkNames

Synopsis

```
void getAllTopLevelTopologicalLinkNames(
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the interface described in 3.3.5 getAllTopLevelTopologicalLinks, page 3-23, but instead of returning entire object structures, this interface returns their names.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of top-level topological links to return in the first batch.
nameList	globaldefs::NamingAttributes List_T	Out	First batch of top-level topological link names.
nameIt	globaldefs::NamingAttributesI terator_I	Out	Iterator to retrieve the remaining top-level topological link names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.3.7 getTopLevelTopologicalLink

Synopsis

```
void getTopLevelTopologicalLink(
        in globaldefs::NamingAttributes_T topoLinkName,
        out topologicalLink::TopologicalLink_T topoLink)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request the top-level topological link if its name is under the control of the EMS.

If a link that is under the control of the EMS is removed from the NE, the ProvType parameter is reported as INVALID.

Parameters

Name	Type	Input/Output	Description
topoLinkName	globaldefs::NamingAttributes_T	In	Name of the topological link.
topoLink	TopologicalLink::TopologicalLink_T	Out	Top-level topological link.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when topoLinkName does not reference a top level topological
link object.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when topoLinkName references a top level topological link
object that does not exist in the EMS.
```

Compliance

TMF-defined.

3.3.8 createTopologicalLink

Synopsis

```
void createTopologicalLink(
        in topologicalLink::TLCreateData_T createData,
        out topologicalLink::TopologicalLink_T topologicalLink)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to create a topological link manually between two MEs. CTM supports topological link creation between MEs belonging to the same multilayer subnetwork. If CTM autodiscovers a topological link between two endpoints on which a manually created link exists, CTM overwrites the manually created topological link information with the autodiscovered topological link information. CTM generates an attribute value change event when a manually created link is overwritten with autodiscovered topological link information. CTM supports topological link creation between two unmanaged NEs, and also between a managed NE and an unmanaged NE. CTM ignores the topological link layer rate if the topological link creation is between two unmanaged NEs.

CTM generates an object creation event notification when a new topological link is created.

Note CTM creates topological links between two PTPs even if there are no existing physical link connections in the network. To enable SDCC on the topological link, the NMS must ensure the presence of physical link connectivity.

Parameters

Name	Type	Input/Output	Description
createData	topologicalLink::TLCreateData_T	In	Data structure containing details for creating new topological links.
topologicalLink	topologicalLink::TopologicalLink_T	Out	Created topological link information.

CTM support for TLCreateData_T structure is as follows:

- string userLabel
  - Supported. The value of this field is assigned to the name of the newly created topological link. For patchcord links, CTM assigns the link name. The user label that is passed as input is ignored for patchcord links.
- boolean forceUniqueness
  - CTM guarantees the uniqueness of topological link names.
- string owner
  - Not supported.
- globaldefs::ConnectionDirection_T direction

  - CTM supports CD_BI for bidirectional links.
  - CTM supports CD_UNI for unidirectional links.
- transmissionParameters::LayerRate_T rate. For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.
- globaldefs::NamingAttributes_T aEndTP
  - Supported.
- globaldefs::NamingAttributes_T zEndTP
  - Supported.
- globaldefs::NVSList_T additionalCreationInfo
  - Supported. The supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support topological links	linkProtectionType	PROT_ONE_PLUS_ONE.
All managed elements that support topological links	linkCost	Integer, 0 to 999999.
All managed elements that support topological links	provType	MANUAL (default value) and PATCHCORD.
All managed elements that support topological links	zEndSecSrc	A string that represents a sequence of namevalue pairs for the TMF physical location of a PTP. A semicolon (;) is used as separator. Format: `name='EMS';value='Cisco Systems/CTM';name='ManagedElement';value='15454-ANSI-68-8';name='PTP';value='/rack=1/shelf=1/slot=15/port=10'`
All managed elements that support topological links	aEndSecDst	A string that represents a sequence of namevalue pairs for the TMF physical location of a PTP. A semicolon (;) is used as separator.

- globaldefs::sSrlgGroupList_T srlgGroupListValue
  - Supported. This field holds the shared risk link group (SRLG) group values. This is an optional parameter. You can include the SRLG group during link creation. A maximum of five SRLG groups are supported for a given link. SRLG groups are only supported for SONET/SDH topological links.

To create unidirectional OTS links without side between 80WXC cards on the ONS15454 NEs, specify the parameter zEndSecSrc as secondary source and the parameter aEndSecDst as destination TP in the additionalCreationInfo list.

```
A End PTP
        [0]    name       = EMS                      value      = Cisco Systems/CTM
        [1]    name       = ManagedElement           value      = 15454-ANSI-68-9
        [2]    name       = PTP                        value
=/rack=1/shelf=1/slot=15/port=12

Z End PTP
        [0]    name       = EMS                      value      = Cisco Systems/CTM
        [1]    name       = ManagedElement           value      = 15454-ANSI-68-8
        [2]    name       = PTP                        value      =
/rack=1/shelf=1/slot=13/port=13

Additional CreationInfo
```

```
        [0]    name       = linkProtectionType        value      = PROT_UNPROTECTED
        [1]    name       = linkCost                  value      = 1024
        [2]    name       = provType                  value      = PATCHCORD
        [3]    name       = zEndSecSrc                value      = name='EMS';value='Cisco
Systems/CTM';name='ManagedElement';value='15454-ANSI-68-8';name='PTP';value='/rack=1/shelf
=1/slot=13/port=10'
        [4]    name       = aEndSecDst                value      = name='EMS';value='Cisco
Systems/CTM';name='ManagedElement';value='15454-ANSI-68-9';name='PTP';value='/rack=1/shelf
=1/slot=15/port=1'
```

In both of these examples, if the operator supplies any wrong parameters for secondary source/drop ports, the CTM Gateway/CORBA interface returns an exception that reports the list of actual available ports retrieved from the ME, such as the following:

```
java.lang.String errorReason=The secondary PTP /rack=1/shelf=1/slot=15/port=10 does not
belong to the list of available PTPs: /rack=1/shelf=1/slot=15/port=1,
/rack=1/shelf=1/slot=15/port=2, /rack=1/shelf=1/slot=15/port=3,
/rack=1/shelf=1/slot=15/port=4, /rack=1/shelf=1/slot=15/port=5,
/rack=1/shelf=1/slot=15/port=6, /rack=1/shelf=1/slot=15/port=7,
/rack=1/shelf=1/slot=15/port=8
```

If you do not provide a secondary source PTP, the primary link between the AendTP and the ZendTP is successfully created. However, while creating the reverse link, the CTM Gateway/CORBA interface tries using the secordary source as the first element from the list of available PTPs returned by the ME/NCP layer (for example, /rack=1/shelf=1/slot=15/port=1). If such a PTP port is still available, the reverse link is created properly; otherwise, CTM throws an exception.

It is recommended that you specify a secondary source PTP. If you do not specify the secondary source PTP, CTM Gateway/CORBA, by default, uses the first item from the list of available PTPs. However, the element may also be invalid.

You have to specify the desired PTP in the additional creation info list in case of the availability of multiple choices.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if input createData is not valid.

EXCPT_INVALID_INPUT - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.3.9 createTopologicalLinkWithYCableProtection

Synopsis

```
void createTopologicalLinkWithYCableProtection(
        in topologicalLink::MTLCreateData_T createData,
        out topologicalLink::TopologicalLink_T topologicalLink)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to create a Y-cable topological link manually between two managed elements. A Y-cable link is a bidirectional link with one PTP source and two PTP destinations. Because of the limitation in TMF for topological links that have one PTP source and one PTP destination, CTM reports two topological links for a Y-cable link. These two topological links have the same nativeEMSName. See 2.2.5.8 Topological Link, page 2-36 for the naming convention for the Y-cable topological link object name. The destination of the Y-cable link must be on a DWDM card on an ONS 15454 SONET/SDH or on an ONS 155xx NE.

CTM generates two object creation event notifications when a new Y-cable topological link is created.

The topological link returned by this method is one leg of the Y-cable.

Parameters

Name	Type	Input/Output	Description
createData	topologicalLink:MTLCreateData_T	In	Data structure containing details for creating a new Y-cable topological link.
topologicalLink	topologicalLink::TopologicalLink_T	Out	One leg of the created Y-cable topological link information.

CTM support for MTLCreateData_T structure is as follows:

- string userLabel
  - Supported. The value of this field is assigned to the nativeEMSName of the newly created topological link.
- boolean forceUniqueness
  - CTM guarantees the uniqueness of topological link names.
- string owner
  - Not supported.
- globaldefs::ConnectionDirection_T direction
  - Supports only CD_BI for bidirectional links.
- transmissionParameters::LayerRate_T rate. Supported layer rates are:
  - LR_Physical_2_5_Gigabit_ITU
  - LR_Physical_10_Gigabit_ITU
  - LR_DSR_OC3_STM1
  - LR_DSR_OC12_STM4
  - LR_DSR_OC48_STM16
  - LR_DSR_OC192_STM64
- globaldefs::NamingAttributesList_T aEndTPs
  - Supported. It contains only one PTP.

- globaldefs::NamingAttributesList_T zEndTPs
  - Supported. It contains two PTPs on the same ME.
- globaldefs::NVSList_T additionalInfo
  - Supported. The supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support topological links	linkCost	Integer, 0 to 999999.
All managed elements that support topological links	provType	MANUAL (default value) and PATCHCORD.

- globaldefs::sSrlgGroupList_T srlgGroupListValue
  - Supported. This field holds the SRLG group values. This is an optional parameter. You can include the SRLG group during link creation. A maximum of five SRLG groups are supported for a given link. SRLG groups are only supported for SONET/SDH topological links.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if input createData is not valid.

EXCPT_INVALID_INPUT - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.3.10 deleteTopologicalLink

Synopsis

```
void deleteTopologicalLink(
        in globaldefs::NamingAttributes_T topoLinkName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to delete a topological link manually between two managed elements. CTM throws the EXCPT_UNABLE_TO_COMPLY exception if an attempt is made to delete an autodiscovered topological link or if the topological link cannot be deleted in the current state. CTM generates an object deletion event notification when a topological link is deleted.

Note

- CTM deletes topological links between two PTPs even if there is a physical link connection in the network. The NMS must ensure the disabling of the SDCC, the deletion of all circuits carried through the link, and the removal of the physical link connectivity.
- Patchcord links are deleted twice. The first delete operation forces the link to become invalid, but the link information is still present in the CTM database. The second delete operation removes the link information from the CTM database.

Parameters

Name	Type	Input/Output	Description
topoLinkName	globaldefs::NamingAttributes_T	In	Name of the topological link to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised if input topological link name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised if the specified topological link name does not exist in
CTM.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_UNABLE_TO_COMPLY - Raised if an attempt is made to delete an auto-discovered
topological link.
```

Compliance

CTM-defined. This interface is already proposed in TMF and will be included TMF Release 3.0.

3.3.11 modifyTopologicalLink

Synopsis

```
void modifyTopologicalLink(
                  in topologicalLink::TLCreateData_T    newTLCreateData,
                  out topologicalLink::TopologicalLink_T newTopologicalLink )
              raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to modify the following parameters of existing topological links:

- Cost attribute (modifiable only for SONET and SDH links)
- SRLG values (modifiable only for SONET and SDH links)

Note The modifyTopologicalLink API is not supported for links with patchcord provisioning. Also, you cannot use the modifyTopologicalLink API to modify the link description on any supported NEs.

Parameters

Name	Type	Input/Output	Description
createData	topologicalLink::TLCreateData_T	In	Data structure containing details for creating a new topological link.
topologicalLink	topologicalLink::TopologicalLink_T	Out	Created topological link information.

CTM support for TLCreateData_T structure is as follows:

- string userLabel
  - Supported. The value of this field is assigned to the name of the newly created topological link.
- boolean forceUniqueness
  - CTM guarantees the uniqueness of topological link names.
- string owner
  - Not supported.
- globaldefs::ConnectionDirection_T direction
  - CTM supports CD_BI for bidirectional links and CD_UNI for unidirectional links.
- transmissionParameters::LayerRate_T rate. For a list of supported layer rates, see the layer rates supported in 3.3.8 createTopologicalLink.
- globaldefs::NamingAttributes_T aEndTP
  - Supported.
- globaldefs::NamingAttributes_T zEndTP
  - Supported.
- globaldefs::NVSList_T additionalCreationInfo
  - Supported. The supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support topological links	linkProtectionType	PROT_ONE_PLUS_ONE.
All managed elements that support topological links	linkCost	Integer, 0 to 999999.

- globaldefs::sSrlgGroupList_T srlgGroupListValue
  - Supported. This field holds the SRLG group values. This is an optional parameter. You can include the SRLG group during link creation. A maximum of five SRLG groups are supported for a given link. SRLG groups are supported only for SONET/SDH topological links.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised if input createData is not valid.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.3.12 getEMS

Synopsis

```
void getEMS(out EMS_T emsInfo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request EMS information.

Parameters

Name	Type	Input/Output	Description
emsInfo	EMS_T	Out	EMS information.

CTM supports EMS_T is as follows:

- globaldefs::NamingAttributes_T name
  - EMS ID is returned as the same name that is displayed on the CTM client EMS property sheet.
- string userLabel
  - Not supported.
- string nativeEMSName
  - CTM returns the EMS ID that is displayed on the CTM client EMS property sheet.
- string owner
  - Not supported.
- string emsVersion
  - CTM returns “9.2.”
- string type
  - CTM returns “Cisco Systems/Cisco Transport Manager.”
- globaldefs::NVSList_T additionalInfo
  - Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

TMF-defined.

3.3.13 getAllL2Topologies

Synopsis

```
void getAllL2Topologies(in multiLayerSubnetwork::L2TopoType_T l2TopoType,
        in unsigned long how_many,
        out multiLayerSubnetwork::L2TopologyList_T l2TopoList,
        out multiLayerSubnetwork::L2TopologyIterator_I l2TopoIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists all L2 topologies of a particular topology type in the EMS. The L2 topology has a scope in the EMS. CTM supports only Resilient Packet Ring (RPR) and point-to-point L2 topology types.

Parameters

Name	Type	Input/Output	Description
l2TopoType	MultiLayerSubnetwork::L2TopoType_T	In	Type of L2 topologies to return.
how_many	unsigned long	In	Maximum number of L2 topologies to return in the first batch.
l2TopoList	multiLayerSubnetwork::L2TopologyList_T	Out	List of available L2 topologies in this multilayer subnetwork.
l2TopoIt	multiLayerSubnetwork::L2TopologyIterator_I	Out	Iterator used to retrieve the remaining L2 topologies.

CTM support for L2Topology_T structure is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. The field contains two tuples. The first tuple contains a value for EMS. The second tuple contains the L2 topology name.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported. The L2 topology name is reported.
- string owner
  - Not supported.
- L2TopoLayerRate_T layerRate
  - Supported. This represents the total capacity of the L2 topology. L2TopoLayerRate_T contains the following information:
    - LayerRate_T layerRate—Represents the layer rate of the SNCs. For VCAT, represents the layer rate of the member SNCs.
    - Unsigned long numberOfVcatMembers—Represents the number of members in each SNC of an L2 topology consisting of all VCAT SNCs. In all other cases, it contains a value less than one and should be ignored.
    - Supported layer rates are:

      LR_STS1_and_AU3_High_Order_VC3

LR_STS3c_and_AU4_VC4

LR_STS6c_and_VC4_2c

LR_STS9c_and_VC4_3c

LR_STS12c_and_VC4_4c

LR_STS24c_and_VC4_8c

LR_Not_Applicable (if the L2 topology contains L1 SNCs with different layer rates or if the L2 topology contains a mix of CCAT and VCAT L1 SNCs)

- multiLayerSubnetwork::L2TopoType l2TopoType
  - The L2 topology type. Valid values are L2TopoType.PT2PT, L2TopoType.RPR, and L2TopoType.DOT17_RPR.
- multiLayerSubnetwork::L2TopoState topoState
  - CTM GateWay/CORBA handles the L2 topology states L2TopoState.COMPLETE and L2TopoState.INCOMPLETE.

Note The L2TopoState API does not take into consideration the L2 protection information (WRAPPING or STEERING). Consequently, the COMPLETE_WRAPPING and COMPLETE_STEERING states are remapped into COMPLETE. Likewise, the INCOMPLETE_WRAPPING and INCOMPLETE_STEERING states are remapped into INCOMPLETE.

- multiLayerSubnetwork::L2TopoOperState
  - The operational state of the L2 topology. Valid values are L2TopoOperState.IS, L2TopoOperState.ISPARTIAL, and L2TopoOperState.OOS.
- multiLayerSubnetwork::L2TopoSyncState topoSyncState
  - The current L2 topology synchronization state. Valid values for L2TopoSyncState are:
    - L2TopoSyncState. NEEDED
    - L2TopoSyncState. INPROGRESS
    - L2TopoSyncState. COMPLETE
    - L2TopoSyncState. PARTIALLY_COMPLETE
    - L2TopoSyncState. FAILED
    - L2TopoSyncState. L2SERVICENOTREADY
- globaldefs::NVSList_T additionalInfo
  - Supported.
  - If the topology type is DOT17_RPR, the bandwidth parameter values return the absolute bandwidth reserved for the class of traffic. The return value is similar to 250 (Class A). The number 250 indicates 250 Mbps bandwidth reserved for Class A-type traffic. Multicast bandwidth parameters are not supported for this type of topology. If no values are defined for class types, the following default values are returned:
    - ClassSpMgmt: C
    - ClassCommitted: C
    - ClassBestEffort: C

- ClassAvvidControl: B
- ClassAvvidVoiceVideo: A
- ClassAReservedBandwidth: 0
- ClassA1Bandwidth: 10
- ClassBCommittedBandwidth: 10
- SpMgmtCoS: 6, 7
- CommittedCoS: 1, 2
- AvvidVoiceVideoCoS: 5
- AvvidControlCoS: 3
- CosCommit: 2

– The bandwidth parameters SpMgmtBandwidth, CommittedBandwidth, BestEffortBandwidth, AvvidControlBandwidth, SpMgmtCoS, CommittedCoS, AvvidVoiceVideoCoS, AvvidControlCoS, and the CoS value CosCommit are supported. Either all or none of these parameters are passed. Each of the bandwidth parameters (SpMgmtBandwidth, CommittedBandwidth, BestEffortBandwidth, and AvvidControlBandwidth) is from 1 to 96; the sum of all four bandwidth parameters is from 1 to 99. The CoS commit value is from 0 to 7. If none of these parameters is passed, CTM takes the following default values:

- SpMgmtBandwidth: 5
- CommittedBandwidth: 80
- BestEffortBandwidth: 10
- AvvidControlBandwidth: 4
- SpMgmtCoS: 6, 7
- CommittedCoS: 1, 2
- AvvidVoiceVideoCoS: 5
- AvvidControlCoS: 3
- CosCommit: 2

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when an l2TopoType is invalid.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.3.14 getAllQoSTemplateNames

Synopsis

```
void getAllQoSTemplateNames( in unsigned long how_many,
        out globaldefs::NamingAttributesList_T qosTempNameList,
        out globaldefs::NamingAttributesIterator_I qosTempNameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists all QoS template names in the EMS. The QoSTemplate has a scope in the EMS.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of qosTempNames to return in the first batch.
qosTempNameList	globaldefs::NamingAttributesList_T	Out	First batch of names.
qosTempNameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator used to retrieve the remaining names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.3.15 getAllQoSTemplates

Synopsis

```
void getAllQoSTemplates
        in unsigned long how_many,
        out QoSTemplateList_T qosTempList
        out QoSTemplateIterator_I qosTempIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists all QoS templates in the EMS. The QoSTemplate has a scope in the EMS.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of qosTempNames to return in the first batch.
qosTempList	QoSTemplateList_T	Out	First batch of QoS templates.
qosTempIt	QoSTemplateIterator_I	Out	Iterator used to retrieve the remaining QoS templates.

CTM support for QoSTemplate structure is as follows:

- globaldefs::NamingAttributes_T templateName—Supported. This field specifies the template name.
- string qos_template_description—Supported. This field specifies the description of the template.
- emsMgr::TemplateType_T—TemplateType specifies the type of template. CTM supports the following types:
  - TemplateType_T.BEST_EFFORT
  - TemplateType_T.CIR_PIR
  - TemplateType_T.ADVANCED
- QOSPolicy_T policy—Supported. This field specifies the QoS policy for the template.
- QOSPolicy_T is defined as follows:
  - string policy_name—Supported. This field specifies the policy name for the QoS.
  - QoSClassList_T qosClass—Supported. This field provides a list of QoS classes for the policy.

    Each QoSClass_T has the following members:

    - multiLayerSubnetwork::MatchProtocolList_T classification—Supported. Each MatchProtocol_T is defined as:
    - Classification_T matchTraffic—Specifies the traffic classification. CTM supports the following classifications:

      Classification_T.MATCH_ANY

      Classification_T.MATCH_DSCP

      Classification_T.MATCH_IP

      Classification_T.MATCH_COS

      string value—This field is a string value. Multiple values are separated by spaces.
- string class_name—Supported. This field specifies the name of the QoS class.
- QoSPolicer_T policer—Supported. This field specifies the policer to use.
- QoSPolicer_T structure has the following members:
  - multiLayerSubnetwork::CIRType_T cir_type—Supported. This field specifies the type of CIR (line rate or rate limited). CTM defines the following values:
  - CIRType_T.CIR_RATE
  - CIRType_T.CIR_LINE
  - long cir—Supported. This field specifies the CIR value in the range of 96 to 800,000 kilobits per second.
  - long cir_burst—Supported. This field specifies the CIR burst value in the range of 8000 to 64000 bytes.
  - multiLayerSubnetwork::COSMarking_T cir_cos—Supported. Specifies the type of CoS marking for the CIR. CTM supports the following values:
  - COSMarking_T.COSMARKING_MARK_COS
  - COSMarking_T.COSMARKING_TRUST
  - long cir_cos_val—Supported. This field specifies the CIR CoS value.

  - multiLayerSubnetwork::ExcessAction_T excess_act—Supported. CTM supports the following values:
  - ExcessAction_T.EXCESS_ACTION_DISCARD
  - ExcessAction_T.EXCESS_ACTION_ALLOW
  - long pir—Supported. This field specifies the PIR value in the range of 96 to 800,000 kilobits per second. The PIR value is equal to or greater than the CIR value.
  - long pir_burst—Supported. This field specifies the PIR burst value in the range of 8000 to 64000 bytes. The PIR burst value is equal to or greater than the CIR burst value.
  - multiLayerSubnetwork::COSMarking_T pir_cos—Supported. This field specifies the type of CoS marking for PIR. CTM supports the following values:
    - COSMarking_T.COSMARKING_MARK_COS
    - COSMarking_T.COSMARKING_TRUST
  - long pir_cos_value—Supported. This field specifies the PIR COS value.
  - multiLayerSubnetwork::ViolateAction_T violate_act—Supported. CTM supports the following values:
  - ViolateAction_T.VIOLATE_ACTION_DISCARD
  - ViolateAction_T.VIOLATE_ACTION_ALLOW
  - long violate_cos_val—Supported. This field specifies the violated COS value.
  - multiLayerSubnetwork::BesteffortType_T beff_type—Supported. CTM supports the following values for best effort type:
    - BestEffortType_T.BEST_EFFORT_RATE
    - BestEffortType_T.BEST_EFFORT_LINE
  - long beff_rate—Supported. This field specifies the maximum rate of best effort in the range of 96 to 800,000 kilobits per second.
  - long beff_rate_burst—Supported. This field is in the range of 8000 to 64000 bytes.
- multiLayerSubnetwork::MatchOption_T operator—Supported. CTM supports the following Match option:
  - MatchOption_T.MATCH_OPTION_OR
  - MatchOption_T.MATCH_OPTION_AND

Note CTM always uses MatchOption_T.MATCH_OPTION_AND.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.3.16 getQoSTemplate

Synopsis

```
void getAllQoSTemplates(
        in globaldefs::NamingAttribute_T QoSTemplateName,
        out QoSTemplate_T qosTemp )
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists the complete QoSTemplate information for each QoSTemplate name. The QoSTemplate has a scope in the EMS.

Parameters

Name	Type	Input/Output	Description
qosTemplateName	globaldefs::NamingAttributes_T	In	QoS template name.
qosTemp	QoSTemplate_T	Out	QoS template with the complete information.

For details about the QoSTemplate_T structure, see 3.3.14 getAllQoSTemplateNames, page 3-38.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when the QoSTemplate name is not valid.

EXCPT_ENTITY_NOT_FOUND - Raised when the QoSTemplate name does not exist on the EMS.
```

Compliance

CTM-defined.

3.3.17 getAllEMSandMEClearedAlarms

Synopsis

```
void getAllEMSandMEClearedAlarms(
        in notifications::ProbableCauseList_T excludeProbCauseList,
        in notifications::PerceivedSeverityList_T excludeSeverityList,
        in unsigned long how_many
        out notifications::EventList_T eventList,
        out notifications::EventIterator_I eventIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the cleared and acknowledged alarms that are under the control of the EMS (both alarms raised by the NEs and alarms raised by the EMS itself). Some alarms may be filtered out (excluded) by specifying their probable causes or severities.

Due to performance concerns, this operation is restricted to enable only one OSS client to access it at one time. Otherwise, CTM throws the EXCPT_UNABLE_TO_COMPLY exception.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

ONS 15310, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH

The ONS 15310, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH do not send a “clear” for TCA events. TCAs are not reported in the eventList that is returned by this method.

ONS 15540 and ONS 15530

TCAs are reported in the eventList that is returned by this method. The ONS 15540 sends a “clear” for TCAs.

Parameters

Name	Type	Input/Output	Description
excludeProbCauseList	notifications::ProbableCause List_T	In	List of probable alarm causes to exclude from the output event list.
excludeSeverityList	notifications::PerceivedSeveri tyList_T	In	List of alarm severities to exclude from the output event list.
how_many	unsigned long	In	Maximum number of alarms to report in the first batch.
EventList	notifications::EventList	Out	First batch of alarms.
EventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_UNABLE_TO_COMPLY - Raised when this method is being used by another OSS Client.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.3.18 getAllEMSSystemClearedAlarms

Synopsis

```
void getAllEMSSystemClearedAlarms(
        in notifications::PerceivedSeverityList_T excludeSeverityList,
        in unsigned long how_many
        out notifications::EventList_T eventList,
        out notifications::EventIterator_I eventIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the cleared and acknowledged alarms that are for the EMS itself. Some alarms may be filtered out (excluded) by specifying their severities.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

Parameters

Name	Type	Input/Output	Description
excludeSeverityList	notifications::PerceivedSeverityList_T	In	List of alarm severities to exclude from the output event list.
how_many	unsigned long	In	Maximum number of alarms to report in the first batch.
EventList	notifications::EventList	Out	First batch of alarms.
EventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.3.19 createServerTrailLink

Synopsis

```
void createServerTrailLink(
        in serverTrailLink::STLCreateData_T createData
        raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to create a server trail link manually between two MEs. CTM supports the server trail link creation between MEs belonging to the same multilayer subnetwork. CTM generates an object creation event notification when a new server trail link is created.

Note

- STLCreateData_T is a CTM-defined data structure. The TMF proposed this structure and included it in the TMF 3.0 specification.
- CTM creates the server trail link between two CTPs even if there are no existing physical link connections in the network.

Parameters

Name	Type	Input/Output	Description
createData	serverTrailLink::STLCreateData_T	In	Data structure containing details for creating a new server trail link.

CTM support for STLCreateData_T structure is as follows:

- string userLabel

  Not supported. The value of this field is assigned to the name of the newly created server trail link.

- boolean forceUniqueness

  Not supported.

- string owner

  Not supported.

- globaldefs::ConnectionDirection_T direction

  CTM supports only CD_BI for bidirectional links.

- transmissionParameters::LayerRate_T rate

  Supported layer rates are:

  - LR_STS1_and_AU3_High_Order_VC3
  - LR_STS3c_and_AU4_VC4
  - LR_STS12c_and_VC4_4c
  - LR_STS24c_and_VC4_8c
  - LR_STS48c_and_VC4_16c
  - LR_STS192c_and_VC4_64c

- globaldefs::NamingAttributes_T aEndCTP

  Supported.

- globaldefs::NamingAttributes_T zEndCTP

  Supported.

- globaldefs::NVSList_T additionalCreationInfo

  Supported. The supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support ServerTrailLink	serverTrailLinkProtectionType	PREEMPTIBLE, UNPROTECTED, FULLY_PROTECTED.
All managed elements that support ServerTrailLink	linkCost	Integer, 0 to 999999.
All managed elements that support ServerTrailLink	linkCount	Integer, 1 to 1024.

- globaldefs::sSrlgGroupList_T srlgGroupListValue

  Supported. This field holds the SRLG group values. This is an optional parameter that you can include in the SRLG group during the link creation. A maximum of five SRLG groups are supported for a given link. SRLG groups are only supported for SONET/SDH server trail links.

CTM support for ServerTrailLink_T structure is as follows:

- globaldefs::NamingAttributes_T name

  Supported. This field has two tuples. The first tuple is for EMS. The second one is for ServerTrailLink. The value in the second tuple is the same as the nativeEMSName field.

- string userLabel

  Not supported.

- boolean forceUniqueness

  Not supported

- string owner

  Not supported.

- globaldefs::ConnectionDirection_T direction

  CTM supports only CD_BI for bidirectional links.

- transmissionParameters::LayerRate_T rate

  Supported layer rates are:

  - LR_STS1_and_AU3_High_Order_VC3
  - LR_STS3c_and_AU4_VC4
  - LR_STS12c_and_VC4_4c
  - LR_STS24c_and_VC4_8c
  - LR_STS48c_and_VC4_16c
  - LR_STS192c_and_VC4_64c

- globaldefs::NamingAttributes_T aEndTP

  Supported. The CTP for the serverTrailLink aEnd.

- globaldefs::NamingAttributes_T zEndTP

  Supported. The CTP for the serverTrailLink zEnd.

- globaldefs::NVSList_T additionalCreationInfo

  Supported. The supported parameters are listed in the following table.

Managed Element Type	Parameter Name	Supported Parameter Values
All managed elements that support ServerTrailLink	serverTrailLinkProtectionType	PREEMPTIBLE, UNPROTECTED, FULLY_PROTECTED.
All managed elements that support ServerTrailLink	linkCost	Integer, 0 to 999999.
All managed elements that support ServerTrailLink	linkCount	Integer, 1 to 1024.

- globaldefs::sSrlgGroupList_T srlgGroupListValue

  Supported. This field holds the SRLG group values. This is an optional parameter that you can include in the SRLG group during the link creation. A maximum of five SRLG groups are supported for a given link. SRLG groups are only supported for SONET/SDH server trail links.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised if input createData is not valid.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.3.20 deleteServerTrailLink

Synopsis

```
void deleteServerTrailLink(
        in globaldefs::NamingAttributes_T serverTrailLinkName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to delete a server trail link manually between two MEs. CTM throws the EXCPT_UNABLE_TO_COMPLY exception if the server trail link cannot be deleted in the current state. CTM generates an object deletion event notification when a server trail link is deleted.

Parameters

Name	Type	Input/Output	Description
serverTrailLinkName	globaldefs::NamingAttributes_T	In	Name of the server trail link to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised if input the server trail link name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised if the specified server trail link name does not exist in
CTM.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_UNABLE_TO_COMPLY - Raised if attempt is made to delete a server trail which cannot
be deleted.
```

Compliance

CTM-defined.

3.3.21 addNetworkPartition

Synopsis

```
void addNetworkPartition(
        in string name, in string description)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the OSS to add a new network partition.

Parameters

Name	Type	Input/Output	Description
name	string	In	Network partition name.
description	string	In	Description of the new partition.

Throws

```
globaldefs::ProcessingFailureException
```

Compliance

CTM-defined.

3.3.22 addMultiLayerSubnetwork

Synopsis

```
void addMultiLayerSubnetwork(in MultiLayerSubnetworkCreateData _T subnetCreateData)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows the NMS to add a new multilayer subnetwork. Upon successful addition, the EMS generates an object creation event for the multilayer subnetwork object.

Parameters

Name	Type	Input/Output	Description
subnetCreateData	MultiLayerSubnetworkCreateData _T	In	MultiLayerSubnetwork input data.

The structure for MultiLayerSubnetworkCreateData _T is as follows:

```
struct MultiLayerSubnetworkCreateData_T
{
        globaldefs::NamingAttributes_T subnetworkName;
        string userLabel;
        string nativeEMSName;
        string owner;
        multiLayerSubnetwork::Topology_T subnetworkType;
        globaldefs::NVSList_T additionalInfo ;
};
```

The description of each parameter is as follows:

Name	Type	Description
subnetworkName	globaldefs::NamingAttributes_T	New subnet partition name.
userLabel	string	User label.
nativeEMSName	string	EMS name.
owner	string	Owner.
subnetworkType	multiLayerSubnetwork::Topology_T	Subnetwork topology.
additionalInfo	globaldefs::NVSList_T	The NMS uses the additionalInfo attribute to communicate to the EMS any other information required to create the multiLayerSubnetwork.

```
enum Topology_T  { TOPO_SINGLETON,  TOPO_CHAIN,  TOPO_PSR,
    TOPO_OPEN_PSR,  TOPO_SPRING,  TOPO_OPEN_SPRING,  TOPO_MESH  };
```

The structure of Additional Info is as follows:

Name	Value	Description	Mandatory or Optional
NETWORK PARTITION	subnetwork name	Name of the network partition where the subnet is added.	Mandatory
Type	subnetwork type	Subnet type.	Mandatory (see the following values)

The parameter name “Type” in additionalInfo can have the following values:

- SONET
- SDH
- OTHER
- UNKNOWNSUBNETTYPE (the default value if a value other than the three preceding values is specified)

Throws

```
globaldefs::ProcessingFailureException
```

Compliance

CTM-defined.

3.3.23 deleteNetworkPartition

Synopsis

```
void deleteNetworkPartition(in string name)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the OSS to delete an existing network partition.

Parameters

Name	Type	Input/Output	Description
name	string	In	Network partition name to delete.

Throws

```
globaldefs::ProcessingFailureException
```

Compliance

CTM-defined.

3.3.24 deleteMultiLayerSubnetwork

Synopsis

```
void deleteMultiLayerSubnetwork(in globaldefs::NamingAttributes_T subnetworkName)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows the NMS to delete an existing subnetwork. Successful operation requires the EMS to generate an object deletion event.

Parameters

Name	Type	Input/Output	Description
subnetworkName	globaldefs::NamingAttributes_T	In	Subnet name to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_ENTITY_NOT_FOUND- Raised if the multilayer subnetwork is not found.

EXCPT_UNABLE_TO_COMPLY- Raised if the EMS cannot comply.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.
```

Compliance

CTM-defined.

3.3.25 getAllSupportedMEModels

Synopsis

```
void getAllSupportedMEModels(out globaldefs::NVSList_T supportedMEs)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of all of the models that the EMS supports. A model's structure has the following form:

- name = "ModelName"
- value = meModel

Parameters

Name	Type	Input/Output	Description
supportedMEs	NVSList_THolder	Out	List of all of the supported model names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.4 EmsSession::EmsSession_I

The emsSession module enables the client to query the EMS to determine the manager interfaces that the EMS supports. The NMS retrieves an instance of the required manager interface objects. This interface is inherited from session::Session_I.

```
interface EmsSession_I:session::Session_I
```

3.4.1 getEventChannel

Synopsis

```
void getEventChannel(
        out CosNotifyChannelAdmin::EventChannel eventChannel)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to access the event channel to receive notifications. CTM publishes the name of the notification channel through the naming service. You can use the naming service to get the corresponding object reference.

Parameters

Name	Type	Input/Output	Description
eventChannel	CosNotifyChannelAdmin::EventChannel	Out	Event channel that the NMS uses.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

TMF-defined.

3.4.2 getManager

Synopsis

```
void getManager(
        in string managerName,
        out common::Common_I managerInterface)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to access the specified manager interface.

Parameters

Name	Type	Input/Output	Description
managerName	string	In	Class or type of manager object that the client wants.
managerInterface	common::Common_I	Out	The actual object returned implements the specified manager interface. However, it is returned as a Common_I object, so the operation can be generic. The client narrows the returned object to the correct object type.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - This exception indicates that the EMS does not support the
manager.

EXCPT_INTERNAL_ERROR - This exception indicates a nonspecific internal EMS failure.
```

Compliance

TMF-defined.

3.4.3 getSupportedManagers

Synopsis

```
void getSupportedManagers(out managerNames_T supportedManagerList)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request the manager interfaces that the EMS implements.

Parameters

Name	Type	Input/Output	Description
supportedManagerList	managerNames_T	Out	List of manager names supported in the form managerName, where *managerName* is one of the following defined manager strings for CTM: • EMS • ManagedElement • MultiLayerSubnetwork • EquipmentInventory • PerformanceManagement • Maintenance • Protection • Software

Throws

None.

Compliance

TMF-defined.

3.5 emsSessionFactory::EmsSessionFactory_I

A single instance of the EmsSessionFactory_I is the entry point to the EMS. The NMS uses this object reference to connect to CTM. This interface implements the version interface and returns the server IDL version. It is inherited from mtnmVersion::Version_I.

```
interface EmsSessionFactory_I:mtnmVersion::Version_I
```

3.5.1 getEmsSession

Synopsis

```
void getEmsSession(in string user,
        in string password,
        in nmsSession::NmsSession_I client,
        out emsSession::EmsSession_I emsSessionInterface)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to obtain the EmsSession_I object, which contains all managers of the EMS.

CTM allows 4 to 25 concurrent login sessions and as many user profiles as required for configuring the number of concurrent login sessions. See B.8 Location of the Naming Service IOR File, page B-6 to configure the number of concurrent login sessions. The username and password must contain from 1 to 53 characters.

This interface can also accept the encrypted username and password. To enable encryption on the username and password, check the **Enable encryption for username and password** check box in the CTM client Control Panel for the CTM GateWay/CORBA service. CTM then decrypts the username and password accordingly.

To encrypt the username and password, you must use the RSA algorithm with the public key or the public key pair provided by CTM. To obtain the CTM public key, use the emsSessionFactory::EmsSessionFactory_I::getEmsPublicKey API. To obtain the CTM public key pair, use the emsSessionFactory::EmsSessionFactory_I::getEmsPublicKeyPair API.

Parameters

Name	Type	Input/Output	Description
user	string	In	User or application that is trying to access the server. This user must have a profile created in CTM.
password	string	In	Password for the user application.
client	nmsSession::NmsSession_I	In	Handle to the NMS NmsSession_I object, to which the returned EmsSession_I object must be associated.
emsSessionInterface	emsSession::EmsSession_I	Out	CORBA Interoperable Object Reference (IOR) for the EmsSession_I interface.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of a nonspecific internal EMS failure.

EXCPT_INVALID_INPUT - Raised when the client is invalid or when the length of the username
or password passed is not within the permissible limits (1-53).

EXCPT_ACCESS_DENIED - Raised in the following cases:

When all the eight login sessions are used up.
When the user name specified does not exist.
Or in case of any other security violation.
```

Compliance

TMF-defined.

3.5.2 getEmsPublicKey

Synopsis

```
void getEmsPublicKey(out string publicKey)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to obtain the string representation of the RSA public key encoded in the Base64 encoding scheme. The operation encrypts the username and password from the string representation of the RSA public key.

The OSS client must use the Base64 decoders to decode the public key and get the byte[] of the public key from the decoded public key string. The byte[] corresponding to the public key represents the key in its primary encoded format; in this case, X.509 SubjectPublicKeyInfo.

You can create the RSA public key using this byte array and cryptographic libraries. Use this key to encrypt the username and password. Bouncy Castle Provider is an example of a security provider that you can use to create the RSA public key.

The public key is then used to encrypt the username and password. Before passing the encrypted password to CTM for login, the OSS client must encode the encrypted username and password using the Base64 encoder to get the string equivalent of the encrypted data.

Parameters

Name	Type	Input/Output	Description
publicKey	string	Out	Public key string returned by CTM.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.5.3 getEmsPublicKeyPair

Synopsis

```
void getEmsPublicKeyPair(out string publicKeyModulus,
        out string publicKeyExponent)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to obtain the string representation of the RSA public key pair (in the modulus and exponent forms) from CTM for encrypting the username and password.

The OSS client must convert the string representation of the public key modulus and public key exponent into a nonnegative integer and create an RSA public key using cryptographic libraries. Crypto++ is one of the security providers available.

Before passing the encrypted password to CTM for login, the OSS client must use the Base64 encoder to encode the encrypted username and password. This provides the string equivalent of the encrypted data.

Parameters

Name	Type	Input/Output	Description
publicKeyModulus	string	Out	Public key modulus string returned by CTM.
publicKeyExponent	string	Out	Public key exponent string returned by CTM.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.6 equipment::EquipmentInventoryMgr_I

This interface accesses operations that deal with equipment. A handle to an instance of this interface is gained through the getManager operation in emsSession. This interface is inherited from common::Common_I.

```
interface EquipmentInventoryMgr_I:common::Common_I
```

This interface implements the getCapabilities method. Operations described in the following sections are returned to the NMS when it invokes getCapabilities.

3.6.1 getAllEquipment

Synopsis

```
void getAllEquipment(in globaldefs::NamingAttributes_T meOrHolderName,
      in unsigned long how_many,
      out EquipmentOrHolderList_T eqList,
      out EquipmentOrHolderIterator_I eqIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the equipment and equipment holders contained in a managed element or equipment holder.

Parameters

Name	Type	Input/Output	Description
meOrHolderName	globaldefs::NamingAttributes_T	In	Name of the ME or equipment holder for which to retrieve contained equipment and equipment holders.
how_many	unsigned long	In	Maximum number of EquipmentOrHolder_Ts to return in the first batch.
eqList	EquipmentOrHolderList_T	Out	First batch of EquipmentOrHolder_Ts.
eqIt	EquipmentOrHolderIterator_I	Out	Iterator used to retrieve the remaining EquipmentOrHolder_Ts.

The EqList parameter contains a sequence of the EquipmentOrHolder_T data type, which is defined as the union of Equipment_T and EquipmentHolder_T.

CTM support for Equipment_T is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field has four tuples. The first tuple is for EMS, the second is for ManagedElement, the third is for EquipmentHolder, and the fourth contains equipment. The value of the fourth tuple is the same as the value in the nativeEMSName field.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported. The naming convention is described in 2.2.5.7 Equipment Holder, page 2-35.
- string owner
  - Not supported.
- boolean alarmReportingIndicator
  - Supported for ONS 15327 and ONS 15454 only.
- ServiceState_T serviceState
  - Not supported.
- EquipmentObjectType_T expectedEquipmentObjectType
  - MGX 8880/8850/8830—All values listed in Table D-1 on page D-2 are supported.
  - ONS 15216—All values listed in Table D-2 on page D-3 are supported.
  - ONS 15305 and ONS 15305 CTC—All values listed in Table D-3 on page D-4 and Table D-4 on page D-4 are supported.
  - ONS 15310 CL—All values listed in Table D-5 on page D-5 are supported.
  - ONS 15310 MA SDH—All values listed in Table D-6 on page D-5 are supported.
  - ONS 15310 MA SONET—All values listed in Table D-7 on page D-5 are supported.
  - ONS 15327—All values listed in Table D-8 on page D-6 are supported.
  - ONS 15454 SDH—All values listed in Table D-9 on page D-6 are supported.
  - ONS 15454 SONET—All values listed in Table D-10 on page D-9 are supported.
  - ONS 15530—All values listed in Table D-11 on page D-13 are supported.
  - ONS 15540—All values listed in Table D-12 on page D-26 and Table D-13 on page D-32 are supported.
  - ONS 15600 SDH—All values listed in Table D-14 on page D-35 are supported.
  - ONS 15600 SONET—All values listed in Table D-15 on page D-35 are supported.
- EquipmentObjectType_T installedEquipmentObjectType
  - Supported. The value reported by the NE is inserted in this field.
- string installedPartNumber
  - Supported when equipment is installed.
- string installedVersion
  - Supported when equipment is installed.

- string installedSerialNumber
  - Supported when equipment is installed.
- globaldefs::NVSList_T additionalInfo
  - Supported when equipment is installed. Common Language Equipment Identification (CLEI) code is available.

CTM support for EquipmentHolder_T is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field has three tuples: one for EMS, one for ManagedElement, and one for EquipmentHolder name.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported. The naming convention is defined in 2.2.5.7 Equipment Holder, page 2-35.
- string owner
  - Not supported.
- boolean alarmReportingIndicator
  - Not supported.
- EquipmentHolderType_T holderType
  - Supported values are additionalHolder, rack, shelf, and slot.
  - For the ONS 15600 SONET and ONS 15600 SDH, valid values are rack, shelf, slot, sub_slot, and ppm_holder.
  - For the ONS 15310 CL, ONS 15310 MA SONET, and ONS 15310 MA SDH, valid values are rack, shelf, slot, and ppm_holder.
  - For the ONS 15540 and ONS 15530, valid values are rack, shelf, slot, sub_slot, and ppm_holder.
  - For the ONS 15454 SONET and ONS 15454 SDH, valid values are rack, shelf, slot, ppm_holder, and port.
  - For all equipment supporting SFP modules, valid values are rack, shelf, slot, sub_slot, and port_holder.
  - For the MGX 8880/8850/8830, valid values are rack, shelf, slot, sub_slot, and peripheral_holder.
- globaldefs::NamingAttributes_T expectedOrInstalledEquipment
  - Supported. If the slot is empty and not preprovisioned for equipment, this field is empty.
- EquipmentObjectTypeList_T acceptableEquipmentTypeList
  - Supported.
  - Not supported for ONS 15530 and ONS 15540.

Table 3-2 Acceptable Equipment for ONS 15454 SONET

Acceptable Equipment	Classic Shelf	M2 Shelf	M6 Shelf
OPT_AMP_17_C, OPT_AMP_L, OPT_BST, OPT_BST_E, OPTBST_L, OPT_PRE, OPT_AMP_C,OPT_RAMP_C, OPT_RAMP_CE, 40-SMR1-C, 40-SMR2-C, TDCU-C, TDCU-F 10GE_XP, GE_XP, 10DME, 2.5G_DM, 2.5G_DMP, ADM_10G, MXP_2.5G_10E, MXP_2.5G_10G, TXPP_MR_2.5G, TXP_MR_2.5G, TXP_MR_10E, TXP_MR_10G, TXP_MR_2.5G,OTU2_XP 32DMX, 32DMX_L, 32DMX_O, 32MUX_O, 32_WSS, 32WSS_L, 4MD, 40DMX, 40MUX, 40WSS, 4MD 40WXC AD_1B, AD_1C, AD_2C, AD_4B, AD_4C MMU, OSC-CSM,PSM 80-WXC-C, 40G_MXP_C	1, 2, 3, 4, 5, 6, 12, 13, 14, 15, 16, 17	2, 3	2, 3, 4, 5, 6, 7
ML1000, ML100T, ML-100X-8, ML-MR-10, CE-MR-10, CE-100T-8, CE-1000-4 (Only when XC10G, XCVXL-10G is installed in slot 8 or 10.)	1, 2, 3, 4, 5, 6, 12, 13, 14, 15, 16, 17	N/A	N/A
DS1, DS3, DS3E, DS3XM, DS3XM_12, DS3I, ETH100, ETH1000, MRC_12, MRC25G_4, OC12, OC3_4, OC12_4, OC3_8, OC48	1, 2, 4, 14, 16, 17	N/A	N/A
DS1, DS3, DS3E, DS3XM, DS3XM_12, DS1N, DS3IN, DS3N, DS3NE, ETH100, ETH1000, MRC_12, MRC25G_4, OC12, OC3_4, OC12_4, OC3_8, OC48	3, 15	N/A	N/A

Table 3-2 Acceptable Equipment for ONS 15454 SONET

Acceptable Equipment	Classic Shelf	M2 Shelf	M6 Shelf
DS1,DS3,DS3E,DS3XM,DS3XM_12,DS3I,ETH100, ETH1000,MRC_12,MRC25G_4,OC12,OC3_4,OC48,OC192, OC192_XFP	5, 16, 12, 13	N/A	N/A
TCC, TCC2, TCC3	7, 11	N/A	N/A
XC10G, OSCM	8, 10	N/A	N/A
AIC, AICI	9	N/A	N/A
TNC, TSC	N/A	1	1, 8

Table 3-3 Acceptable Equipment for ONS 15454 SDH

Acceptable Equipment	Classic Shelf	M2 Shelf	M6 Shelf
OPT_AMP_17_C, OPT_AMP_L, OPT_BST, OPT_BST_E, OPTBST_L, OPT_PRE, OPT_AMP_C,OPT_RAMP_C, OPT_RAMP_CE, 40-SMR1-C, 40-SMR2-C, TDCU-C, TDCU-F 10GE_XP, GE_XP, 10DME, 2.5G_DM, 2.5G_DMP, ADM_10G, MXP_2.5G_10E, MXP_2.5G_10G, TXPP_MR_2.5G, TXP_MR_2.5G, TXP_MR_10E, TXP_MR_10G, TXP_MR_2.5G,OTU2_XP 32DMX, 32DMX_L, 32DMX_O, 32MUX_O, 32_WSS, 32WSS_L, 4MD, 40DMX, 40MUX, 40WSS, 4MD 40WXC AD_1B, AD_1C, AD_2C, AD_4B, AD_4C MMU, OSC-CSM,PSM 80-WXC-C, 40G_MXP_C	1, 2, 3, 4, 5, 6, 12, 13, 14, 15, 16, 17	2, 3	2, 3, 4, 5, 6, 7
ML1000, ML100T, ML-100X-8, ML-MR-10, CE-MR-10, CE-100T-8, CE-1000-4 (Only when XC10G, XCVXL-10G is installed in slot 8 or 10.)	1, 2, 3, 4, 5, 6, 12, 13, 14, 15, 16, 17	N/A	N/A

Table 3-3 Acceptable Equipment for ONS 15454 SDH

DS1, DS3, DS3E, DS3XM, DS3XM_12, DS3I, ETH100, ETH1000, MRC_12, MRC25G_4, OC12, OC3_4, OC12_4, OC3_8, OC48	1, 2, 4, 14, 16, 17	N/A	N/A
E3, DS3IN, E1_42, STM1E_12, STM1_8, STM4_4, ETH100, ETH1000, MRC12, MRC25G_12, STM1_4, STM16, STM4	3, 15	N/A	N/A
DS1, DS3, DS3E, DS3XM, DS3XM_12, DS1N, DS3IN, DS3N, DS3NE, ETH100, ETH1000, MRC_12, MRC25G_4, OC12, OC3_4, OC12_4, OC3_8, OC48	3, 15	N/A	N/A
DS31, E3, ETH100, ETH1000,MRC_12,MRC25G_12, STM1_4, STM16, STM4, STM64, STM64_XFP	5, 13	N/A	N/A
ETH100, ETH1000, MRC12, MRC25G_12, STM1_4, STM16, STM4, STM64, STM64_XFP	6, 12	N/A	N/A
TCC, TCC2, TCC3	7, 11	N/A	N/A
XCVXC_10G, XC10G, XCVXL_10G, XCVXL_2.5, XCVXL_10G, OSCM	8, 10	N/A	N/A
AICI	9	N/A	N/A
TNC, TSC	N/A	1	1, 8

ONS 15216 FlexLayer Slot Number	Acceptable Equipment
1, 2, 3, 4	FLA-8-*channel ID* *Channel ID* = 36.6, 44.5, 52.5, 60.6
	FLB-2-*channel ID* *Channel ID* = 31.1, 32.6, 35.0, 36.6, 38.9, 40.5, 42.9, 44.5, 46.9, 48.5, 50.9, 52.5, 54.9, 56.5, 58.9, 60.6
	CS-2
	CS-3
	CS-4
	CS-SM-Y
	CS-MM-Y
	SC-4B
1, 2, 3	VOA-4

ONS 15216 DCU R2.0 Slot Number	Acceptable Equipment
1, 2	DCU-100 (100 ps/nm)
1, 2	DCU-350 (350 ps/nm)
1, 2	DCU-450 (450 ps/nm)
1, 2	DCU-550 (550 ps/nm)x
1, 2	DCU-750 (750 ps/nm)
1, 2	DCU-950(ps/nm)
1, 2	DCU-1150 (1150 ps/nm)
1, 2	DCU-E-200 (200 ps/nm)
1, 2	DCU-E-350 (350 ps/nm)
1, 2	DCU-L-300 (300 ps/nm)
1, 2	DCU-L-600 (600 ps/nm)
1, 2	DCU-L-700 (700 ps/nm)
1, 2	DCU-L-800 (800 ps/nm)
1, 2	DCU-L-1000 (1000 ps/nm)
1, 2	DCU-L-1100 (1100 ps/nm)
1, 2	DCU-DS-L-100 (100 ps/nm)
1, 2	DCU-DS-L-200 (200 ps/nm)
1, 2	DCU-DS-L-300 (300 ps/nm)

ONS 15305 and ONS 15305 CTC Slot Number	Acceptable Equipment
1, 2, 3, 4	All equipment listed in D.4 Equipment List for ONS 15305, page D-4

ONS 15310 CL Slot Number	Acceptable Equipment
1	CTX-2500, CTX-FILLER
2	CE-100T-8, ML-100T-8, CE-MR-6

ONS 15310 MA SDH Slot Number	Acceptable Equipment
3, 4	CTX-2500, CTX-FILLER
1, 2, 5, 6	E1-21/E3-DS3-3, E1-63/E3-DS3-3
1, 2, 5, 6	CE-100T-8, ML-100T-8, CE-MR-6, EXP-FILLER

ONS 15310 MA SONETSlot Number	Acceptable Equipment
3, 4	CTX-2500, CTX-FILLER
1, 2, 5, 6	DS1-84/DS3-EC1-3, DS1-28/DS3-EC1-3, DS3-EC-6
1, 2, 5, 6	CE-100T-8, ML-100T-8, CE-MR-6, EXP-FILLER

ONS 15327 Slot Number (All Releases)	Acceptable Equipment
7, 8	MIC
5, 6	XTC
1, 2, 3, 4	ETH100, OC12, OC48

ONS 15600 SDH Slot Number	Acceptable Equipment
5,10	TXC
6/7, 8/9	SSXC
1, 2, 3, 4, 11, 12, 13, 14	OC48_16, OC192_4, OC48_32
1, 2, 3, 4, 11, 12, 13, 14	ASAP, STM64_4_DWDM

ONS 15600 SONET Slot Number	Acceptable Equipment
5, 10	TXC
6/7, 8/9	SSXC
1, 2, 3, 4, 11, 12, 13, 14	OC48_16, OC192_4, OC48_32
1, 2, 3, 4, 11, 12, 13, 14	ASAP, OC192_4_DWDM

Compliance

TMF-defined.

3.6.2 getAllEquipmentNames

Synopsis

```
void getAllEquipmentNames(in globaldefs::NamingAttributes_T meOrHolderName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the interface described in 3.6.1 getAllEquipment, page 3-55, but returns the object name instead of returning the entire object.

Parameters

Name	Type	Input/Output	Description
meOrHolderName	globaldefs::NamingAttributes_T	In	Name of the ME or equipment holder for which to retrieve contained equipment and equipment holders.
how_many	unsigned long	In	Maximum number of EquipmentOrHolder_Ts to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when meOrHolderName does not reference a managed element or
an equipment holder.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementOrEquipmentName references an object
that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.6.3 getAllSupportedPTPs

Synopsis

```
void getAllSupportedPTPs(
        in globaldefs::NamingAttributes_T equipmentName,
        in unsigned long how_many,
        out terminationPoint::TerminationPointList_T tpList,
        out terminationPoint::TerminationPointIterator_I tpIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request the PTPs that are directly supported by specific equipment. The PTPs that are returned are those that share their physical layer with the primary equipment (that is, they represent a port on the equipment or are connected by a fiber, wire, and so on).

If the equipment is protection equipment in a 1_FOR_N protection group, CTM reports only the AlarmReporting for the LayeredParameters of the supported PTPs.

- ONS 15454 SONET: CTM returns OC-*n*, DS1, DS3, EC-1, fibre channel, and front Ethernet ports.
- ONS 15454 SDH: CTM returns STM-*n* (OC-*n*), E1, E3, DS3I, STM1E, fibre channel, and front Ethernet ports.
- ONS 15327: CTM returns OC-*n*, DS1, DS3, and Ethernet ports.
- ONS 15540: CTM returns client, wave, wavepatch, filter, wdm, filterband, filtergroup, oscfilter, wavepassthrough, fastEthernet, TenGigEthernetPhy, and sonet-sdh as potential PTPs.
- ONS 15530: CTM returns all the supported PTPs for ONS 15530.
- ONS 15216: CTM returns all the supported PTPs for ONS 15216, if there is a topological link associated with the NE.
- ONS 15600 SONET: CTM returns OC-*n* ports.
- ONS 15600 SDH: CTM returns STM-*n* ports.
- ONS 15305 and ONS 15305 CTC: CTM returns WAN, SDH, and PDH ports.
- ONS 15310 CL and ONS 15310 MA: CTM returns OC-*n*, DS1, DS3, and Ethernet ports.
- MGX 8880/8850/8830: CTM returns the physical service module cards ports (DS1, DS3, and SONET/SDH).

Parameters

Name	Type	Input/Output	Description
equipmentName	globaldefs::NamingAttributes_T	In	Name of the equipment for which to retrieve supported PTPs.
how_many	unsigned long	In	Maximum number of PTPs to report in the first batch.
tpList	terminationPoint::TerminationPointList_T	Out	First batch of PTPs.
tpIt	terminationPoint::TerminationPointIterator_I	Out	Iterator used to retrieve the remaining PTPs.

CTM support for TerminationPoint_T structure is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field contains three tuples for PTP and four tuples for CTP as defined by the TMF. The third tuple contains positional information for the PTP. The naming convention is described in 2.2.5.3 Physical Termination Point, page 2-30, 2.2.5.4 Connection Termination Point, page 2-31, and 2.2.5.5 Floating Termination Point, page 2-35.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported.

- string owner
  - Not supported.
- globaldefs::NamingAttributes_T ingressTrafficDescriptorName
  - Not supported.
- globaldefs::NamingAttributes_T egressTrafficDescriptorName
  - Not supported.
- TPType_T type
  - Supported. CTM supports TPT_PTP, TPT_CTP, and TPT_FTP.
- TPConnectionState_T connectionState
  - Supported. For PTP, it is TPCS_NA; for CTP and floating termination point (FTP), it is TPCS_BI_CONNECTED.
- TerminationMode_T tpMappingMode
  - Not supported.
- Directionality_T direction
  - Supported. PTPs always report D_BIDIRECTIONAL. For the ONS 155xx, D_SOURCE and D_SINK are also supported.
- transmissionParameters::LayeredParameterList_T transmissionParams
  - Supported. Layer rate is supported for PTP, CTP, and FTP.

CTM supports the following parameters for PTP.

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
AlarmReporting			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327 ONS 15454 SONET ONS 15454 SDH ONS 15600 SONET ONS 15600 SDH	On, Off	All equipment (except equipment that does not have PTP, such as XC, TCC2, CXC, and so on).
ServiceState			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327 ONS 15454 SONET ONS 15454 SDH ONS 15600 SONET ONS 15600 SDH	—	All equipment (except ML and CE cards, FCMR, and the equipment that does not have PTP, such as XC, TCC2, and so on).

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
AdminState			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327 ONS 15454 SONET	IN_SERVICE, OUT_OF_SERVICE, OUT_OF_SERVICE_BY_MAINTENANCE (for NE R3.4.1 or later), AUTO_IN_SERVICE (for NE R3.4.1 or later) Down, Up	All equipment (except ML and CE cards, FCMR, and equipment that does not have PTP, such as XC, TCC2, and so on). G1000_4, G1000_2, and FCMR do not support the AUTO_IN_SERVICE value. Down and Up are valid values shown for ML-series cards. For CE-series and E-series Ethernet cards, valid values are In Service, Auto In Service, Out of Service, and Out of Service by Maintenance. SERV_NA is not a valid value for any card.
	ONS 15600	IN_SERVICE, OUT_OF_SERVICE_BY_MAINTENANCE, OUT_OF_SERVICE (for NE R5.0 or later), AUTO_IN_SERVICE (for NE R5.0 or later)	All equipment (except equipment that does not have PTP, such as CXC and so on).
	ONS 15454 SDH ONS 15600 SDH	UNLOCKED LOCKED, DISABLED LOCKED, MAINTENANCE UNLOCKED, AUTO_IN_SERVICE	All equipment (except ML and CE cards, and equipment that does not have PTP, such as XC, TCC2, and so on). FCMR does not support the UNLOCKED, AUTO_IN_SERVICE value. The ONS 15600 SDH does not support LOCKED, DISABLED and UNLOCKED, AUTO IN SERVICE.
LineCode			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327	ami, b8zs	DS1 port on XTC and CTX cards.
	ONS 15454 SONET	ami, b8zs	DS1, DS1N.
	ONS 15310 CL ONS 15310 MA ONS 15454 SONET	b3zs	DS3E, DS3NE, DS3XM, DS3I, DS3IN, DS3, DS1-84/DS3-EC1-3, DS1-28/DS3-EC1-3, ONS 15310 CL and ONS 15310 MA DS3 ports on CTX cards.
	ONS 15454 SDH	hdb3	E1, E1N, E1_42.
	ONS 15454 SDH	b3zs	DS3I, DS31N.

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
FrameFormat			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327	esf, d4, unframed, auto provision fmt (only on ONS 15310)	DS1 port on XTC cards.
	ONS 15454 SONET	esf, d4, unframed	DS1, DS1N.
	ONS 15454 SONET	m13, cbit, unframed, auto provision fmt	DS3E, DS3NE, DS3I, DS31N, DS3.
	ONS 15454 SONET	m13, cbit	DS3XM.
	ONS 15454 SDH	e1_mf, e1_crcmf, e1_unframed	E1, E1N, E1_42.
	ONS 15310 CL ONS 15310 MA	m13, cbit, unframed	DS3 port on CTX cards.
	ONS 15454 SDH	m13, cbit, unframed, auto provision fmt	DS3I, DS31N.
SDH_SONET_SS_BITS			
PTP	ONS 15454 SONET	SDH, SONET	OC-*n* equipment.
	ONS 15454 SDH	SDH	STM-*n* equipment.
EnableSyncMsg			
PTP	ONS 15310 CL ONS 15310 MA ONS 15327 ONS 15454 SONET ONS 15600 SONET	On, Off	OC-*n* equipment.
	ONS 15454 SDH ONS 15600 SDH,	On, Off	STM-*n* equipment.
ALSMode			
PTP	ONS 15454 SONET	DISABLED, AUTO, MANUAL, MANUAL_RESTART_FOR_TEST	OC3_8, OC48 ELR, OC192 equipment only.
	ONS 15454 SDH	DISABLED, AUTO, MANUAL, MANUAL_RESTART_FOR_TEST	STM1_8, STM16 ELR, STM64 equipment only.
PulseWidth			
PTP	ONS 15454 SONET	Float value from 2.0 to 100.0	OC3_8, OC48 ELR, OC192 equipment only.
	ONS 15454 SDH	Float value from 2.0 to 100.0	STM1_8, STM16 ELR, STM64 equipment only.
RecoveryInterval			
PTP	ONS 15454 SONET	Integer value from 100 to 300	OC3_8, OC48 ELR, OC192 equipment only.
	ONS 15454 SDH	Integer value from 100 to 300	STM1_8, STM16 ELR, STM64 equipment only.

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
HasLaserBeenShutdown			
PTP	ONS 15454 SONET	True, False (read-only value)	OC3_8, OC48 ELR, OC192 equipment only.
	ONS 15454 SDH	True, False (read-only value)	STM1_8, STM16 ELR, STM64 equipment only.
RestartLaser			
PTP	ONS 15454 SONET	True, False (write-only value)	OC3_8, OC48 ELR, OC192 equipment only.
	ONS 15454 SDH	True, False (write-only value)	STM1_8, STM16 ELR, STM64 equipment only.
IPPMMonitor			
CTP	ONS 15310 CL ONS 15310 MA ONS 15327 ONS 15454 SONET ONS 15600 SONET ONS 15600 SDH	On, Off	STS CTPs on OC-*n* cards.
	ONS 15454 SDH	On, Off	VC4 CTPs on STM-*n* cards.
TrailTraceActualTx			
CTP	ONS 15454 SONET	Free-format string	DS3E, DS3NE, DS3XM, DS3I, DS31N.
	ONS 15454 SDH	Free-format string	E3, DS3I, DS3IN, STM1E_12.
TrailTraceExpectedRx			
CTP	ONS 15310 CL ONS 15310 MA ONS 15327	Free-format string	OC3.
	ONS 15454 SONET	Free-format string	DS3E, DS3NE, DS3I, DS3IN, DS3XM, EC1, OC3, OC3_8, OC12_4, OC48AS, OC192.
	ONS 15454 SDH	Free-format string	E3, DS3I, DS3IN, STM1E_12, OC3, OC3_8, OC12_4, OC48AS, OC192.
	ONS 15600 SONET ONS 15600 SDH	Free-format string	OC48_16, OC192_4.
FTP	ONS 15327	Free-format string	G1000_2.
	ONS 15310 CL ONS 15310 MA	Free-format string	ML2_MAPPER and ML2_L2L3.
	ONS 15454 SONET ONS 15454 SDH	Free-format string	G1000_4, ML-series, and CE-series cards (except G1000; for all other cards it is supported only for R5.0 and later).

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
TrailTraceMonitor			
CTP	ONS 15310 CL ONS 15310 MA ONS 15327	On, Off	OC3.
	ONS 15454 SONET	On, Off	DS3E, DS3NE, DS3I, DS3IN, DS3XM, EC1, OC3, OC3_8, OC12_4, OC48AS, OC192.
	ONS 15454 SDH	On, Off	E3, DS3I, DS3IN, STM1E_12, OC3, OC3_8, OC12_4, OC48AS, OC192.
	ONS 15600 SONET ONS 15600 SDH	On, Off	OC48_16, OC192_4.
FTP	ONS 15327	On, Off	G1000_2.
	ONS 15310 CL ONS 15310 MA	On, Off	ML2_L2L3 and ML2_MAPPER.
	ONS 15454 SONET ONS 15454 SDH	On, Off	G1000_4, ML-series, and CE-series cards (except G1000; for all other cards it is supported only for R5.0 and later).
XCLoopBack			
CTP	ONS 15327 ONS 15454 SONET ONS 15600 SONET ONS 15600 SDH	On, Off	STS CTPs on OC-*n* cards when line loopback is not set on the containing PTP.
	ONS 15454 SDH	On, Off	VC4 CTPs on STM-*n* cards when line loopback is not set on the containing PTP.
PortMediaType			
PTP	ONS 15454 SONET ONS 15454 SDH	UNDEFINED, FibreChannel-1 Gbps, FibreChannel-2 Gbps, FICON-1 Gbps, and FICON-2 Gbps	FCMR (for NE R5.0 and later).
Link Recovery			
PTP	ONS 15454 SONET ONS 15454 SDH	On, Off	FCMR (for NE R5.0 and later).
No. of Superblocks			
PTP	ONS 15454 SONET ONS 15454 SDH	From 4 to 32	FCMR (for NE R5.0 and later).
Enable Distance Extension			
PTP	ONS 15454 SONET ONS 15454 SDH	On, Off	FCMR (only in distance extension mode and for NE R5.0 and later).

TP Type	NE Type	Valid Values	Supporting Equipment and Restrictions
Auto Detect Credits			
PTP	ONS 15454 SONET ONS 15454 SDH	On, Off	FCMR (only in distance extension mode and for NE R5.0 and later).
Number Credits			
PTP	ONS 15454 SONET, ONS 15454 SDH	From 2 to 256, in multiples of 2	FCMR (only in distance extension mode and for NE R5.0 and later).
Autoadjust GFP Buffer Threshold			
PTP	ONS 15454 SONET ONS 15454 SDH	On, Off	FCMR (only in distance extension mode and for NE R5.0 and later).
GFP Buffers Available			
PTP	ONS 15454 SONET ONS 15454 SDH	From 16 to 1200, in multiples of 16	FCMR (only in distance extension mode and for NE R5.0 and later).
Ingress Idle Filtering			
PTP	ONS 15454 SONET ONS 15454 SDH	On, Off	FCMR (only in distance extension mode and for NE R5.0 and later).
MaxFrameSize			
PTP	ONS 15454	From 2148 to 2172	FCMR (only in distance extension mode).
FrameFormat			
CTP	ONS 15454	Auto provision fmt, esf, d4, and unframed	DS3XM_12.
FDLMode			
CTP	ONS 15454	T1.403 and BFDL	DS3XM_12.
DetectedLineType			
PTP	ONS 15454	Read-only value	DS3E.

Note If TrailTraceMonitor is off for a given CTP or FTP, CTM does not return TrailTraceActualRx in transmissionParams.

- TPProtectionAssociation_T tpProtectionAssociation
  - Not supported.
- boolean edgePoint
  - Supported. For CTP and FTP, the value must be False.
- globaldefs::NVSList_T additionalInfo
  - Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when equipmentName does not reference an equipment object.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentName references an object that does not
exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.6.4 getAllSupportedPTPNames

Synopsis

```
void getAllSupportedPTPNames(
        in globaldefs::NamingAttributes_T equipmentName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the interface described in 3.6.3 getAllSupportedPTPs, page 3-63, but returns the object names instead of returning the entire object.

Parameters

Name	Type	Input/Output	Description
equipmentName	globaldefs::NamingAttributes_T	In	Name of the equipment for which to retrieve supported PTPs.
how_many	unsigned long	In	Maximum number of PTPs to report in the first batch.
nameList	obaldefs::NamingAttributesList_T	Out	First batch of PTP names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining PTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentName does not reference an equipment object.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentName references an object that does not
exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.6.5 getAllSupportingEquipment

Synopsis

```
void getAllSupportingEquipment(
        in globaldefs::NamingAttributes_T ptpName,
        out EquipmentOrHolderList_T eqList)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request the equipment that contains a given PTP.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP for which to retrieve supporting equipment.
eqList	EquipmentOrHolderList_T	Out	List of equipment (not equipment holders) that directly implements the PTP.

EquipmentOrHolderList_T is a list of EquipmentOrHolder_T, which is defined as a union of Equipment_T and EquipmentHolder_T structures. For details about both structures, see 3.6.1 getAllEquipment, page 3-55.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when ptpName does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when ptpName references a PTP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.6.6 getAllSupportingEquipmentNames

Synopsis

```
void getAllSupportingEquipmentNames(
        in globaldefs::NamingAttributes_T ptpName,
        out globaldefs::NamingAttributesList_T nameList)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.6.5 getAllSupportingEquipment, page 3-72, but returns the object name instead of returning the entire object.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP for which to retrieve the supporting equipment.
nameList	globaldefs::NamingAttributesList_T	Out	List of equipment names (not equipment holders) directly implementing the PTP.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when ptpName does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when ptpName references a PTP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.6.7 getContainedEquipment

Synopsis

```
void getContainedEquipment(
        in globaldefs::NamingAttributes_T equipmentHolderName,
        out EquipmentOrHolderList_T equipmentOrHolderList)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the equipment and equipment holders that a specific equipment holder contains. This interface differs from getAllEquipment in that it looks only at the next level of the containment hierarchy.

Parameters

Name	Type	Input/Output	Description
equipmentHolderName	globaldefs::NamingAttributes_T	In	Name of the equipment holder for which to retrieve the directly contained equipment and equipment holders. CTM also accepts the ME name as a value for this parameter.
equipmentOrHolderList	EquipmentOrHolderList_T	Out	Directly contained equipment and equipment holders.

EquipmentOrHolderList_T is a list of EquipmentOrHolder_T, which is defined as a union of Equipment_T and EquipmentHolder_T structures. For details about both structures, see 3.6.1 getAllEquipment, page 3-55.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentHolderName does not reference an equipment
holder object.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentHolderName references an equipment holder
that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.6.8 getEquipment

Synopsis

```
void getEquipment(in globaldefs::NamingAttributes_T equipmentOrHolderName,
        out EquipmentOrHolder_T equip)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the equipment or equipment holder for specific equipment or equipment holder names.

Parameters

Name	Type	Input/Output	Description
equipmentOrHolderName	globaldefs::NamingAttributes_T	In	Name of the equipment or equipment holder to retrieve.
equip	EquipmentOrHolder_T	Out	Returned equipment or equipment holder.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentOrHolderName does not reference an equipment or
an equipment holder.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentOrHolderName references an equipment or
equipment holder that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.6.9 provisionEquipment

Synopsis

```
void provisionEquipment(
        in EQTCreateData_T equipmentCreateData,
        out Equipment_T createdEquipment)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to permanently provision a piece of equipment in an equipment holder in an ME. The result of this operation is the creation of the equipment object and all of its related objects, such as TPs. The resulting object is returned.

The equipment holder that contains the equipment must exist. The equipment object can already exist, but the equipment holder must not have expected equipment already provisioned.

If the equipment is provisioned successfully, CTM generates a state change event on the holder state change and an object creation event for the equipment, and pushes them to the event channel.

It is important to note that the ONS 15454 SONET detects new equipment and provisions it automatically, provided that the equipment holder (slot) is empty and is not preprovisioned for another module type.

The equipment must be from the acceptableEquipmentTypeList of the slots. If these slots are selected to provision any other type of card, CTM returns the EXCPT_UNABLE_TO_COMPLY exception.

PPMs and PIMs can also be provisioned using this interface.

It is not necessary to know the subtypes of the OC-n card at the time of provisioning. When equipment is actually installed, CTM reports the subtype in the name field of the equipment.

For the ONS 15600 SONET and ONS 15600 SDH, each CXC or SSXC card occupies two slots (slots 6 and 7 or slots 8 and 9), and the NMS can specify any one of the slots to provision a CXC or SSXC card.

Parameters

Name	Type	Input/Output	Description
equipmentCreateData	EQTCreateData_T	In	Data that describes the equipment to create.
createdEquipment	Equipment_T	Out	Resulting created equipment.

CTM support for EQTCreateData_T structure is as follows:

- string userLabel
  - Not supported.
- boolean forceUniqueness
  - Not supported.
- string owner
  - Not supported.
- EquipmentObjectType_T expectedEquipmentObjectType
  - This parameter must take one of the values for equipment described in 3.6.1 getAllEquipment, page 3-55.
- globaldefs::NamingAttributes_T equipmentHolderName

  - Supported.
- globaldefs::NVSList_T additionalInfo
  - Not supported.

Note This interface is not applicable for configuring payloads for client ports in TXP_MR_10G.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_OBJECT_IN_USE - Raised when the equipment holder already has an expected equipment.

EXCPT_INVALID_INPUT - Raised when equipmentHolder does not reference an equipmentHolder
object.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentHolder references an object that does not
exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the equipment cannot be created at the NE.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.6.10 setAlarmReportingOff

Synopsis

```
void setAlarmReportingOff(
        in globaldefs::NamingAttributes_T equipmentOrHolderName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface disables alarm reporting on a piece of equipment. If successful, CTM sends a warning alarm notification. For optical NEs, the interface supports the node names as equipment only when the NE alarms are configured with "off."

Parameters

Name	Type	Input/Output	Description
equipmentOrHolderName	Globaldefs::NamingAttributes_T	In	Equipment on which to deactivate alarm reporting.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentOrHolderName does not reference an equipment or
holder.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when equipmentOrHolderName references an object that does
not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when alarm reporting cannot be disabled for the equipment
or holder.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.6.11 setAlarmReportingOn

Synopsis

```
void setAlarmReportingOn(
        in globaldefs::NamingAttributes_T equipmentOrHolderName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables alarm reporting on a piece of equipment.

The default setting for alarmReporting is “on.” If the alarmReporting status is off, it indicates that a warning alarm has been raised for it. CTM generates a cleared warning alarm if successful. For optical NEs, the interface supports the node names as equipment only when the NE alarms are configured with "off."

Parameters

Name	Type	Input/Output	Description
equipmentOrHolderName	globaldefs::NamingAttributes_T	In	Equipment on which to activate alarm reporting.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentOrHolderName does not reference an equipment or
holder.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentOrHolderName references an object that does
not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when alarm reporting cannot be disabled for the equipment
or holder.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.6.12 unprovisionEquipment

Synopsis

```
void unprovisionEquipment(
        in globaldefs::NamingAttributes_T equipmentName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to permanently unprovision equipment from the managed element. The successful result of this operation is the potential deletion of the equipment object and all of its related objects, such as termination points.

The ONS 15454 SONET does not allow a user to unprovision a piece of equipment that is in use (SNC is created on this equipment). However, if the equipment is present in the slot and is not in use, it can be unprovisioned. If the equipment is not physically removed from the slot and new equipment is not provisioned in the same slot, the NE redetects the equipment in approximately 1 minute. If the equipment is not physically removed from the slot and new equipment of a different type is provisioned in the same slot, the NE generates an equipment mismatch alarm.

If the equipment is unprovisioned successfully, CTM generates a state change event on the holder state change and an object deletion event for the equipment, and pushes them to the event channel.

The ONS 15454 reserves slots 7 and 11 for TCC. One of these slots has an in-service TCC card. If the NMS invokes unprovisionEquipment on this equipment, CTM throws the EXCPT_UNABLE_TO_COMPLY exception.

Parameters

Name	Type	Input/Output	Description
equipmentName	globaldefs::NamingAttributes_T	In	Name of the equipment to unprovision.

Note This interface is not applicable for configuring payloads for client ports in TXP_MR_10G.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when equipmentName does not reference an equipment object.

EXCPT_OBJECT_IN_USE - Deletion is rejected because equipment resources are in use.

EXCPT_ENTITY_NOT_FOUND - Raised when equipmentName references an object that does not
exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the equipment cannot be unprovisioned at the NE.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.7 maintenanceOps::MaintenanceMgr_I

MaintenanceMgr_I is used as a handle to gain access to the maintenance operation. A handle to an instance of this interface is gained through the getManager operation in emsSession.

```
interface MaintenanceMgr_I:common::Common_I
```

This interface also implements the getCapabilities method. Operations described in the following subsections are returned to the NMS when it invokes getCapabilities.

3.7.1 getActiveMaintenanceOperations

Synopsis

```
void getActiveMaintenanceOperations(
        in globaldefs::NamingAttributes_T tpOrMeName,
        in unsigned long how_many,
        out CurrentMaintenanceOperationList_T
            currentMaintenanceOpeationList,
        out CurrentMaintenanceOperationIterator_I cmoIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to query the EMS to determine if any persistent maintenance commands have been invoked. This query is supported for PTP, CTP, and ME objects.

Parameters

Name	Type	Input/Output	Description
tpOrMeName	globaldefs::NamingAttributes_T	In	Name of the PTP, CTP, or ME.
how_many	unsigned long	In	Maximum number of maintenance operations to return in the first batch.
currentMaintenanceOpeationList	CurrentMaintenanceOperationList_T	Out	First batch of maintenance operations.
cmoIt	CurrentMaintenanceOperationIterator_I	Out	Iterator to access the remaining maintenance operations.

CTM support for CurrentMaintenanceOperation structure is as follows:

- globaldefs::NamingAttributes_T tpName
  - Supported.
- MaintenanceOperation_T maintenanceOperation
  - CTM supports FACILITY_LOOPBACK, TERMINAL_LOOPBACK, and PAYLOAD_LOOPBACK.
- transmissionParameters::LayerRate_T layerRate
  - Supported for CTPs only.
- globaldefs::NVSList_T additionalInfo
  - Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpOrMeName does not reference a valid object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpOrMeName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.7.2 performMaintenanceOperation

Synopsis

```
void performMaintenanceOperation(
        in CurrentMaintenanceOperation_T maintenanceOperation,
        in MaintenanceOperationMode_T maintenanceOperationMode)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to operate and release the maintenance commands that are supported by a TP.

ONS 15310 CL and ONS 15310 MA

CTM supports only terminal and facility loopback operations for electrical and OC-n PTPs.

CTM does not support loopback operations for CTPs.

ONS 15327

CTM supports only terminal and facility loopback operations for DS-n and OC-n PTPs.

CTM does not support loopback operations for CTPs.

For the ONS 15327 R3.4 or later, the loopback state can be set only if the TP state is either Out of Service–Maintenance or Auto in Service.

ONS 15454 SDH

CTM supports only terminal and facility loopback operations for PTPs.

CTM does not support loopback operations for CTPs.

For the ONS 15454 SDH R3.4 or later, the loopback state can be set only if the TP state is either Out of Service–Maintenance or Auto in Service.

ONS 15454 SONET

CTM supports only terminal and facility loopback operations for DS-n, OC-n, EC1, 10G-transponder, 10G-muxponder, and G1000_4 PTPs.

Only facility loopback is supported for DS1 CTPs on DS3XM cards.

For the ONS 15454 SONET R3.4 or later, the loopback state can be set only if the TP state is either Out of Service–Maintenance or Auto in Service.

ONS 15600 SDH

CTM supports only terminal and facility loopback operations for PTPs.

CTM does not support loopback operations for CTPs.

Loopback is supported only in the Out of Service–Maintenance state of the PTP.

CTM ignores the MaintenanceOperation_T value in CurrentMaintenanceOperation_T structure, if MaintenanceOperationMode_T is set to MOM_RELEASE.

ONS 15600 SONET

CTM supports only facility and payload loopback operations for PTPs.

CTM does not support loopback operations for CTPs.

Loopback is supported only in the Out of Service–Maintenance state of the PTP.

CTM ignores the MaintenanceOperation_T value in CurrentMaintenanceOperation_T structure, if MaintenanceOperationMode_T is set to MOM_RELEASE.

Parameters

Name	Type	Input/Output	Description
maintenanceOperation	CurrentMaintenanceOperation_T	In	Information on the maintenance operation to perform. This structure contains the field maintenance operation. Valid values for this field are: • FACILITY_LOOPBACK • TERMINAL_LOOPBACK • PAYLOAD_LOOPBACK
maintenanceOperation Mode	MaintenanceOperationMode_T	In	Whether the maintenance operation is completed or released.

CTM support for CurrentMaintenanceOperation_T structure is as follows:

- globaldefs::NamingAttributes_T tpName
  - Supported.
- MaintenanceOperation_T maintenanceOperation
  - Only FACILITY_LOOPBACK, TERMINAL_LOOPBACK, and PAYLOAD_LOOPBACK are supported.
- transmissionParameters::LayerRate_T layerRate
  - Supported for CTPs only.
- globaldefs::NVSList_T additionalInfo
  - Not supported.

Throws

`globaldefs::ProcessingFailureException`

```
EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a TP.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the operation is denied by the ME.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8 managedElementManager::ManagedElementManager_I

This interface accesses operations that deal with MEs and TPs. A handle to an instance of this interface is gained through the getManager operation in emsSession.

```
interface ManagedElementManager_I:common::Common_I
```

This interface also implements the getCapabilities method. Operations described in the following sections are returned to the NMS when it invokes getCapabilities.

3.8.1 getAllActiveAlarms

Synopsis

```
void getAllActiveAlarms(
        in globaldefs::NamingAttributes_T meName,
        in notifications::ProbableCauseList_T excludeProbCauseList,
        in notifications::PerceivedSeverityList_T excludeSeverityList,
        in unsigned long how_many,
        out notifications::EventList_T eventList,
        out notifications::EventIterator_I eventIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all active alarms for the specified ME, excluding alarms that the ME does not report to CTM. Some alarms can be filtered out (excluded) by specifying their probable causes or severities.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

ONS 15310 CL, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH

The ONS 15310 CL, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH do not send a “clear” for TCA events. TCAs are not reported in the eventList that is returned by this method.

ONS 15530 and ONS 15540

TCAs are reported in the eventList returned by this method. The ONS 15530 and ONS 15540 send a “clear” for TCAs.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the ME for which to retrieve alarms.
excludeProbCauseList	notifications::ProbableCauseList_T	In	List of probable causes to exclude (for which events are not reported).
excludeSeverityList	notifications::PerceivedSeverityList_T	In	List of severities to exclude from the output event list.
how_many	unsigned long	In	Maximum number of alarms to report in the first batch.
eventList	notifications::EventList_T	Out	First batch of alarms.
eventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when meName does not reference a managed element object or
excludeProbCauseList contains undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an ME object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.2 getAllSNCs

Synopsis

```
void getAllSNCs(
        in globaldefs::NamingAttributes_T managedElementName,
        in transmissionParameters::LayerRateList_T connectionRateList,
         in unsigned long how_many,
         out subnetworkConnection::SubnetworkConnectionList_T sncList,
         out subnetworkConnection::SNCIterator_I sncIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all SNCs that originate on, terminate on, or transit through a given ME.

- For the ONS 15310 CL, this method does not report STS-1 SNC on CTX using DS1 ports.
- For the ONS 15310 MA SONET, this method does not report STS-1 SNC on DS1 ports of DS1_xx cards.
- For the ONS 15310 MA SDH, this method does not report VC4 SNC on E1 ports of E1_63_E3_3 cards.
- For the ONS 15327, this method does not report STS-1 SNC on XTC using DS1 ports.

- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report VCAT SNCs.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME whose SNCs are required.
connectionLayerRateList	transmissionParameters::LayerRateList_T	In	List of rates of the SNCs to report. If an empty list is specified, all SNCs of all rates are reported.
how_many	unsigned long	In	Maximum number of SNCs to report in the first batch.
sncList	subnetworkConnection::SubnetworkConnectionList_T	Out	First batch of SNCs.
sncIt	subnetworkConnection::SNCIterator_I	Out	Iterator to retrieve the remaining SNCs.

CTM support for SubnetworkConnection_T structure is as follows:

- globaldefs::NamingAttributes_T name
  - Supported. This field contains two tuples. The first tuple contains values for the EMS. The second tuple contains the SNC name, which is the same as nativeEMSName.
- string userLabel
  - Supported.
- string nativeEMSName
  - Supported. The name supplied by the user at creation time is reported.
- string owner
  - Not supported.
- SNCState_T sncState
  - CTM supports SNCS_NONEXISTENT, SNCS_ACTIVE, SNCS_DELETING, and SNCS_PARTIAL.

Note When the SNC is not in SNCS_ACTIVE state, this structure might not have complete information about the SNC. For example, transmissionParameters for TPs reported might not be initialized. When the SNC goes into SNCS_ACTIVE state, a state change event is generated (see A.4.4 State Change Event, page A-6). Then, the NMS invokes the multiLayerSubnetwork::MultiLayerSubnetworkMgr_I::getSNC method to retrieve complete information about the SNC.

- globaldefs::ConnectionDirection_T direction
  - Supported.
- transmissionParameters::LayerRate_T rate
  - Supported.

- StaticProtectionLevel_T staticProtectionLevel
  - Supported. For PCA circuits, the value is set to PREEMPTIBLE.
- SNCType_T sncType
  - Not supported.
- TPDataList_T aEnd
  - Supported.
- TPDataList_T zEnd
  - Supported.
- Reroute_T rerouteAllowed
  - Not supported.
- NetworkRouted_T networkRouted
  - Not supported.
- globaldefs::NVSList_T additionalInfo
  - Supported. Customer ID and service ID are reported if they are available. For an SNC on Ethernet equipment, CTM reports all associated VLAN IDs. For a WDM SNC, the sncSize is reported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an ME object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.3 getAllManagedElements

Synopsis

```
void getAllManagedElements(
        in unsigned long how_many,
        out managedElement::ManagedElementList_T meList,
        out managedElement::ManagedElementIterator_I meIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request details about all MEs that are under the control of ManagedElementMgr_I.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of MEs to report in the first batch.
meList	managedElement::ManagedElementList_T	Out	First batch of MEs.
meIt	managedElement::ManagedElementIterator_I	Out	Iterator to retrieve the remaining MEs.

CTM support for ManagedElement_T structure is as follows:

- globaldefs::NamingAttributes_T name
  - Supported.
- string userLabel
  - Not supported.
- string nativeEMSName
  - Supported. The name is reported by the NE.
- string owner
  - Not supported.
- string location
  - Supported.
- string version
  - Supported.
- string productName
  - Supported.
- CommunicationState_T communicationState
  - Supported.
- boolean emsInSyncState
  - Supported.
- transmissionParameters::LayerRateList_T supportedRates

  For a list of all supported layer rates, see 2.2.1 Layer Rate, page 2-23.
- globaldefs::NVSList_T additionalInfo

  Supported. Two parameters are sent in this field:
  - A name/value pair for the IP address with the name “IPAddress.”
  - A name/value pair for the operational state with the name “OperationalState,” containing the following values. The emsInSyncState value is determined by the OperationalState value. A state change event is generated whenever the state of emsInSyncState changes.

OperationStates	EmsInSyncState
PreProvisioned	False
InServiceSyncConfiguration	False
OutOfService	False

OperationStates	EmsInSyncState
UnderMaintenance	False
InServiceIntializing	False
InService	True

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.4 getAllManagedElementNames

Synopsis

```
void getAllManagedElementNames(
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as 3.8.3 getAllManagedElements, page 3-85, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum number of MEs to report in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of MEs.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining MEs.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.5 getAllPTPs

Synopsis

```
void getAllPTPs(
        in globaldefs::NamingAttributes_T managedElementName,
        in transmissionParameters::LayerRateList_T tpLayerRateList,
        in transmissionParameters::LayerRateList_T connectionLayerRateList,
        in unsigned long how_many,
        out terminationPoint::TerminationPointList_T tpList,
        out terminationPoint::TerminationPointIterator_I tpIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows an NMS to request all PTPs on a specific ME that contain one or more NMS-specified PTP layer rates, and that are capable of supporting one or more NMS-specified connection layer rates. If no PTPs match the layer constraints, an empty list is returned. A PTP is returned regardless of connectivity to other MEs or position in the subnetwork. (Both PTPs at the subnetwork edge and PTPs that are internal to the subnetwork are reported.)

For PTPs on protection equipment in a 1_FOR_N protection group, CTM reports only the AlarmReporting for the LayeredParameters of the PTPs.

This interface also returns a list of front Ethernet ports on an ML-series card on the specified managed element.

- ONS 15216: CTM returns all the supported PTPs for ONS 15216.
- ONS 15305 and ONS 15305 CTC: CTM returns WAN, SDH, and PDH ports.
- ONS 15310 CL, ONS 15310 MA SDH, and ONS 15310 MA SONET: CTM returns OC-*n*, DS1, DS3, and Ethernet ports.
- ONS 15327: CTM returns OC-*n*, DS1, DS3, and Ethernet ports.
- ONS 15454 SONET: CTM returns OC-*n*, DS1, DS3, EC-1, fibre channel, and Ethernet ports.
- ONS 15454 SDH: CTM returns STM-*n* (OC-*n*), E1, E3, DS3I, STM1E, fibre channel, and Ethernet ports.
- ONS 15530: CTM returns all supported PTPs.
- ONS 15540: CTM returns client, wave, wavepatch, filter, wdm, filterband, filtergroup, oscfilter, wavepassthrough, fastEthernet, TenGigEthernetPhy, and sonet-sdh as potential PTPs.
- ONS 15600 SONET: CTM returns OC-*n* ports.
- ONS 15600 SDH: CTM returns STM-*n* ports.
- Unmanaged NE: CTM returns PTPs only when a topological link exists, and returns only those PTPs that participate in the topological link.
- Storage (MDS 9000) NE: CTM returns PTPs only when a topological link exists, and returns only those PTPs that participate in the topological link.
- MGX 8880/8850/8830: CTM returns the physical ports (DS1, DS3, and SONET/SDH) of service module cards.

Parameters

Name	Type	Input/Output	Description
managedElement Name	globaldefs::NamingAttributes_T	In	Name of the ME for which to retrieve PTPs.
tpLayerRateList	transmissionParameters::LayerRateList_T	In	List of PTP layer rates for which the PTPs are retrieved. To be reported, a PTP must contain at least one of the layer rates specified. If the list is empty, all PTPs (of all rates) are returned.
connectionLayer RateList	transmissionParameters::LayerRateList_T	In	CTM does not use this parameter.
how_many	unsigned long	In	Maximum number of PTPs to report in the first batch.
tpList	terminationPoint::TerminationPointList_T	Out	First batch of PTPs.
tpIt	terminationPoint::TerminationPointIterator_I	Out	Iterator to retrieve remaining PTPs.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

For a list of all supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element
object.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME object that does
not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.6 getAllPTPNames

Synopsis

```
void getAllPTPNames(
        in globaldefs::NamingAttributes_T managedElementName,
        in transmissionParameters::LayerRateList_T tpLayerRateList,
        in transmissionParameters::LayerRateList_T connectionLayerRateList,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the interface described in 3.8.5 getAllPTPs, page 3-88, but returns the object names instead of the object structures.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME for which to retrieve PTPs.
tpLayerRateList	transmissionParameters::LayerRateList_T	In	List of PTP layer rates for which PTPs are retrieved. To be reported, a PTP must contain at least one of the specified layer rates. If the list is empty, all PTPs (of all rates) are returned.
connectionLayerRateList	transmissionParameters::LayerRateList_T	In	CTM does not use this parameter.
how_many	unsigned long	In	Maximum number of PTPs to report in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of PTP names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining PTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element
object.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME object that does
not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.7 getContainedInUseTPs

Synopsis

```
void getContainedInUseTPs(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T layerRateList,
        in unsigned long how_many,
        out terminationPoint::TerminationPointList_T tpList,
        out terminationPoint::TerminationPointIterator_I tpIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the in-use CTPs contained in a specific PTP, at specific layer rates. An in-use CTP is a CTP used by an SNC in any state (including pending) or a CTP that is terminated and mapped.

Example for SONET: The NMS invokes this method on an OC3 PTP (and assumes that the NE does not have an XCVT card, so VT-level cross-connects are not possible) with an empty layerRateList.

- If there are no SNCs created, CTM returns zero CTPs.
- If there is one SNC created at the STS-1 level, CTM returns one CTP at the STS-1 layer rate.

Example for SDH: The NMS invokes this method on an STM-1 PTP (and assumes that the NE does not have an XC10G card, so VC12-level cross-connects are not possible) with an empty layerRateList.

- If there are no SNCs created, CTM returns zero CTPs.
- If there is one SNC created at the VC4 level, CTM returns one CTP at the VC4 layer rate.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the PTP.
layerRateList	transmissionParameters::LayerRateList_T	In	List of rates of the contained actual CTPs to return. An empty list tells the EMS to retrieve all contained actual CTPs (of all rates).
how_many	unsigned long	In	Maximum number of CTPs to return in the first batch.
tpList	terminationPoint::TerminationPointList_T	Out	First batch of contained in-use CTPs.
tpIt	terminationPoint::TerminationPointIterator_I	Out	Iterator to retrieve the remaining contained in-use CTPs.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

For a list of all supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a PTP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.8 getContainedInUseTPNames

Synopsis

```
void getContainedInUseTPNames(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T layerRateList,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.8.7 getContainedInUseTPs, page 3-90, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the PTP.
layerRateList	transmissionParameters::LayerRateList_T	In	List of rates of the contained actual CTPs to return. An empty list indicates to the EMS to get all contained actual CTPs (for all rates).
how_many	unsigned long	In	Maximum number of CTPs to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of contained in-use CTP names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining contained in-use CTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a PTP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.9 getContainedPotentialTPs

Synopsis

```
void getContainedPotentialTPs(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T layerRateList,
        in unsigned long how_many,
```

```
    out terminationPoint::TerminationPointList_T tpList,
    out terminationPoint::TerminationPointIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the CTPs it can support in all possible mapping configurations, at specified rates and TPs. This interface returns all potential contained CTPs for a specific TP. The TP might be a PTP or a CTP. If the layerRateList is empty, contained CTPs at all flexible and static LayerRates are returned.

For example, on an ONS 15454 SONET NE, the NMS invokes this method on an OC-3 PTP (assuming that the NE does not have an XCVT card, so VT-level cross-connects are not possible) with an empty layerRateList. Regardless of the number of SNCs, CTM returns four CTPs: three at the STS-1 layer rate and one at the STS3c layer rate.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the PTP or CTP for which to retrieve contained CTPs.
layerRateList	transmissionParameters::LayerRateList_T	In	List of rates of the contained CTPs to report. An empty list tells the EMS to report all contained CTPs (of all rates).
how_many	unsigned long	In	Maximum number of contained CTPs to report in the first batch.
tpList	terminationPoint::TerminationPointList_T	Out	First batch of contained in-use CTPs.
tpIt	terminationPoint::TerminationPointIterator_I	Out	Iterator to retrieve the remaining contained in-use CTPs.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

For a list of all supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a PTP or CTP object.

EXCPT_TP_INVALID_ENDPOINT - Raised when tpName refers to a PTP (and all contained
potential CTPs) that is part of a topological link.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a PTP or CTP object that does not
exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.10 getContainedPotentialTPNames

Synopsis

```
void getContainedPotentialTPNames(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T layerRateList,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
raises(globaldefs::ProcessingFailureException);
```

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the PTP or CTP for which to retrieve contained CTPs.
layerRateList	transmissionParameters::LayerRateList_T	In	List of the rates of the contained CTPs to report. An empty list indicates to the EMS to report all contained CTPs (of all rates).
how_many	unsigned long	In	Maximum number of contained CTPs to report in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of contained in-use CTP names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining contained in-use CTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a PTP or CTP object.

EXCPT_TP_INVALID_ENDPOINT - Raised when tpName refers to a PTP (and all contained
potential CTPs) that is part of a topological link.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a PTP or CTP object that does not
exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.8.11 getContainingSubnetworkNames

Synopsis

```
void getContainingSubnetworkNames(
        in globaldefs::NamingAttributes_T managedElementName,
        out globaldefs::NamingAttributesList_T subnetNames)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of subnetwork names to which the ME (supplied as an input parameter) belongs.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME for which to retrieve the containing subnetwork names.
subnetNames	globaldefs::NamingAttributesList_T	Out	Names of the subnetworks to which the NE belongs.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element
object.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME object that does
not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8.12 getManagedElement

Synopsis

```
void getManagedElement(
        in globaldefs::NamingAttributes_T managedElementName,
        out managedElement::ManagedElement_T me)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the managed element for a specific managed element name.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME to retrieve.
me	managedElement::ManagedElement_T	Out	Retrieved ME.

For details about the ManagedElement_T structure, see 3.8.3 getAllManagedElements, page 3-85.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element
object.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME object that does
not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8.13 getAllCtmMEInternalPatchcord

Synopsis

```
void getAllCtmMEInternalPatchcord(
          in globaldefs::NamingAttributes_T managedElementName,
          out globaldefs::NVSList_T patches)
          raises (globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of internal CTM patchcords for a specific managed element name.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME to retrieve.
patches	globaldefs::NVSList_T	Out	The retrieved patchcords.

For details about the ManagedElement_T structure, see 3.8.3 getAllManagedElements, page 3-85.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when managedElementName does not reference a managed element
object.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME object that does
not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.14 setAdditionalInfo

Synopsis

```
void setAdditionalInfo(
in globaldefs::NamingAttributes_T tpName,
```

```
    inout globaldefs::NVSList_T  additionalInfo)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface sets attributes using the additionalInfo structure. This interface is supported only for:

- Physical termination point (PTP)
- Cards and NE types listed in the following table

Card	ONS 15310 CL	ONS 15310 MA SONET	ONS 15310 MA SDH	ONS 15454 SONET	ONS 15454 SDH
G1000_4				X	X
CE-1000-4				X	X
CE-100T-8	X	X	X	X	X
CE-MR-10				X	X
CE-MR-6		X	X		

This interface is also supported for the attribute LinkIntegrityTimer for the cards and the NE types listed in the following table.

Card	ONS 15310 CL	ONS 15310 MA SONET	ONS 15310 MA SDH	ONS 15454 SONET	ONS 15454 SDH
CE-100T-8	X	X	X	X	X
CE-MR-10				X	X
CE-MR-6		X	X		

This interface is also supported for the attribute LinkIntegrityTimer for the cards and the NE types listed in the following table.

Card	ONS 15454 SONET	ONS 15454 SDH
G1000_4	X	X
CE-1000-4	X	X

For CE-100T-8, CE-MR-6, and CE-MR-10 cards, the LinkIntegrityTimer attribute value cannot be set to 0.

For G1000_4 and CE-1000-4 cards, the LinkIntegrityTimer attribute value cannot be set to 0 if the LinkIntegrityDisable attribute is set to false.

For G1000_4 and CE-1000-4 cards, the LinkIntegrityTimer attribute value must be set to 0 if the LinkIntegrityDisable attribute is set to true.

For G1000_4 and CE-1000-4 cards, the LinkIntegrityDisable attribute can be set only to true or false.

This operation returns a list of modified attributes with the new related values applied.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the TP to which to apply the list of modified attribute values.
additionalInfo	globaldefs::NVSList_T	In/out	List of attributes that must be modified, as well as the new related values to apply.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service, or if the service
is not supported for the selected NE type or card.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a termination point object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a TP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8.15 getTP

Synopsis

```
void getTP(
        in globaldefs::NamingAttributes_T tpName,
        out terminationPoint::TerminationPoint_T tp)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the TP structure for a specific TP name.

For the PTP on protection equipment in a 1_FOR_N protection group, CTM reports only the AlarmReporting for the LayeredParameters of that PTP.

For PTPs, the additionalInfo attribute list contains the LinkIntegrityTimer attribute value if the selected card is one of the following (see also 3.8.14 setAdditionalInfo, page 3-96):

- CE-100T-8 or CE-MR-6 on ONS 15310 MA SONET or ONS 15310 MA SDH
- CE-100T-8 on ONS 15310 CL
- CE-100T-8 or CE-MR-10 on ONS 15454 SONET or ONS 15454 SDH

For PTPs, the additionalInfo attribute list contains the LinkIntegrityTimer and LinkIntegrityDisable attribute values if the selected card is one of following:

- CE-1000-4 and G1000_4 on ONS 15454 SONET or ONS 15454 SDH

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of the TP to retrieve.
tp	terminationPoint::TerminationPoint_T	Out	Retrieved TP.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a termination point object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references a TP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8.16 getAssociatedTopologicalLinks

Synopsis

```
void getAssociatedTopologicalLinks(
        in globaldefs::NamingAttributes_T ptpName,
        out topologicalLink::TopologicalLink_T topologicalLinks)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to pass the TP name. CTM supports multilayer topological links and returns a list of topological links through the TP.

CTM reports all the physical optical and manual topological links.

Parameters

Name	Type	Input/ Output	Description
tpName	globaldefs::NamingAttributes_T	In	Name of TP whose topological link is required.
topologicalLinks	topologicalLink::TopologicalLink_T	Out	Topological link structure whose end is a specific TP.

For details about the TopologicalLink_T structure, see 3.3.5 getAllTopLevelTopologicalLinks, page 3-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when ptpName does not reference a physical termination point
object.

EXCPT_ENTITY_NOT_FOUND - Raised when ptpName references a PTP object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.17 setTPData

Synopsis

```
void setTPData(
        in subnetworkConnection::TPData_T tpInfo,
        out terminationPoint::TerminationPoint_T modifiedTP)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows the NMS to set parameters on a specific TP. The results tell the NMS which modifications succeeded.

CTM supports the following parameters for CTPs:

- IPPMMonitor
- XCLoopBack
- TrailTraceActualTx
- TrailTraceExpectedRx
- TrailTraceMonitor

To completely remove the path trace, the NMS must set TrailTraceMonitor to Off and TrailTraceExpectedRx to an empty string in one call.

If the NMS sends only TrailTraceExpectedRx, CTM sets TrailTraceMonitor to On and sets the value of the expected string. If TrailTraceExpectedRx contains an empty string as the only parameter in the call, CTM changes TrailTraceMonitor to On and sets the expected string to an empty string. To completely remove the path trace, the NMS must set TrailTraceMonitor to Off and set TrailTraceExpectedRx to an empty string in one call.

CTM supports the following parameters for PTPs:

- AlarmReporting
- AdminState
- LineCode
- FrameFormat
- SDH_SONET_SS_BITS
- EnableSyncMsg

The SDH_SONET_SS_BITS parameter can be modified only when EnableSyncMsg is Off. EnableSyncMsg can be set to On only if SDH_SONET_SS_BITS is set to SONET.

The SDH_SONET_SS_BITS and EnableSyncMsg parameters must be modified by invoking two separate method calls. If you try to use the same method to modify SDH_SONET_SS_BITS and EnableSyncMsg, CTM generates a processing failure exception.

The setTPData interface does not apply to the following ML-series card parameters that are configurable through Cisco IOS:

- Admin state, MTU size, and COS accounting for POS and Ethernet interfaces.
- Speed, duplex, and flow control for Ethernet interfaces.

CTM supports the following automatic laser shutdown parameters on PTPs. These parameters are not defined by the TMF.

- ALSMode
- PulseWidth
- RecoveryInterval
- RestartLaser

If the equipment is the protecting equipment in a 1_For_N protection group, CTM sets the PTPs on the equipment to AlarmReporting.

CTM supports only the following parameters for FTPs:

- TrailTraceExpectedRx
- TrailTraceMonitor

For valid values for each parameter, see Table 3-4 in 3.6.3 getAllSupportedPTPs, page 3-63.

You must use the CTM client to set other TP parameters.

Invoking this method from the NMS is a synchronous call.

Parameters

Name	Type	Input/Output	Description
tpInfo	subnetworkConnection::TPData_T	In	Details about required modifications.
modifiedTP	terminationPoint::TerminationPoint_T	Out	Result of modification.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

CTM support for TPData_T structure is as follows:

- globaldefs::NamingAttributes_T tpName
  - Supported.
- terminationPoint::TerminationMode_T tpMappingMode
  - Not supported.
- transmissionParameters::LayeredParameterList_T transmissionParams
  - Supported.
- globaldefs::NamingAttributes_T ingressTrafficDescriptorName
  - Not supported.
- globaldefs::NamingAttributes_T egressTrafficDescriptorName
  - Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the TP referred to in tpInfo does not exist.

EXCPT_INVALID_INPUT - Raised when tpInfo does not reference a valid CTP object.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.8.18 getAllSrcPTPsForSNCProvisioning

Synopsis

```
void getAllSrcPTPsForSNCProvisioning(
        in globaldefs::NamingAttributes_T managedElementName,
        in transmissionParameters::LayerRate_T layerRate,
        in globaldefs::ConnectionDirection_T direction,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T tpList,
        out globaldefs::NamingAttributesIterator_I tpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of PTP names that can be used as the source of an SNC to create on a given ME. The NMS must specify the layer rate and the direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a PTP name list based on current resource availability. The actual SNC creation might still fail.

Parameters

Name	Type	Input/Output	Description
managedElement Name	globaldefs::NamingAttributes_T	In	Name of the ME that is the source of the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. All PTP names available as the source of the SNC at this layer rate are returned.
direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_UNI (unidirectional) or CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of PTP names to report in the first batch.
tpList	globaldefs::NamingAttributesList_T	Out	First batch of PTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining PTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.19 getAllSrcCTPsForSNCProvisioning

Synopsis

```
void getAllSrcCTPsForSNCProvisioning(
        in globaldefs::NamingAttributes_T ptpName,
        in transmissionParameters::LayerRate_T layerRate,
        in globaldefs::ConnectionDirection_T direction,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T tpList,
        out globaldefs::NamingAttributesIterator_I tpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of CTP names that can be used as the source of an SNC to create on a given ME. The NMS must specify the layer rate and the direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a CTP name list based on current resource availability. The actual SNC creation might still fail. Values returned by this list can be used as aEnd in the SNCCreateData_T parameter of the createAndActivateSNCFromUserLabel method described in 3.9.16 createAndActivateSNCFromUserLabel, page 3-138.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP that is the source of the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. All CTP names available at this layer rate are returned.
direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_UNI (unidirectional) or CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of CTP names to report in the first batch.
tpList	globaldefs::NamingAttributesList_T	Out	First batch of CTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining CTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.20 getAllSrcCTPsForVCATMemberSNCProvisioning

Synopsis

```
void getAllSrcCTPsForVCATMemberSNCProvisioning(
    in globaldefs::NamingAttributes_T ptpName,
    in transmissionParameters::LayerRate_T layerRate,
    in globaldefs::ConnectionDirection_T direction,
    in unsigned long how_many,
    out globaldefs:: NamingAttributesList_T tpList,
    out globaldefs:: NamingAttributesIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of CTP names that can be used as the source of a VCAT member SNC to create on a given PTP. The NMS must specify the layer rate and the direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a CTP name list based on current resource availability. The actual SNC creation might still fail. Values returned by this list can be used as the aEnd in each SNCCreateData_T listed from the sncCreateDataList field of the type VCAT SNCCreateData_T, which is required as a parameter of the createVCATl method described in 3.9.26 createVCAT, page 3-156.

This interface must be used according to 2.2.7.2.4 SNC Provisioning for ONS 15454 CE-MR-10 Cards, page 2-41.

Restrict the use of this API to VCAT circuits involving CE-MR-10 cards in Manual mode.

Parameters

See the Parameters section of 3.8.19 getAllSrcCTPsForSNCProvisioning, page 3-103.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.21 getAllDestPTPsForSNCProvisioning

Synopsis

```
void getAllDestPTPsForSNCProvisioning(
    in globaldefs::NamingAttributes_T managedElementName,
    in globaldefs::NamingAttributes_T srcCTPName,
    in transmissionParameters::LayerRate_T layerRate,
    in globaldefs::ConnectionDirection_T direction,
    in unsigned long how_many,
    out globaldefs:: NamingAttributesList_T tpList,
    out globaldefs:: NamingAttributesIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of PTP names that can be used as the destination of an SNC to create on a given ME. The NMS must specify the source CTP name, layer rate, and the direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a PTP name list based on current resource availability. The actual SNC creation might still fail.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the managed element that is the destination of the SNC to create.
srcCTPName	globaldefs::NamingAttributes_T	In	Name of the CTP that is the source of the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. This value must match the layer rate of the source CTP specified in the srcCTPName field. This parameter returns all PTP names that are available as the destination of the SNC at this layer rate.
Direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_UNI (unidirectional) or CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of PTP names to report in the first batch.
tpList	globaldefs::NamingAttributesList_T	Out	First batch of PTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining PTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the PTP does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.22 getAllDestCTPsForSNCProvisioning

Synopsis

```
void getAllDestCTP namesForSNCProvisioning(
        in globaldefs::NamingAttributes_T ptpName,
        in globaldefs::NamingAttributes_T srcCTPName,
        in transmissionParameters::LayerRate_T layerRate,
        in globaldefs::ConnectionDirection_T direction,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T ctpList,
        out globaldefs::NamingAttributesIterator_I tpIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of CTP names that can be used as the destination of an SNC to create on a given PTP. The NMS must specify the source CTP name (or source FTP name, in the case of VC3 port grouping circuits that have the source on E3, DS3I, or DS3IN cards), layer rate, and the direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a CTP list based on current resource availability. The actual SNC creation might still fail. Values returned by this list can be used as the zEnd in the SNCCreateData_T parameter of 3.9.16 createAndActivateSNCFromUserLabel, page 3-138.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP.
srcCTPName	globaldefs::NamingAttributes_T	In	Name of the CTP selected as the source CTP name of the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. This value must match the layer rate of srcCTP specified in the srcCTPName field. All CTP names available at this layer rate are returned.
direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_UNI (unidirectional) or CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of CTP names to report in the first batch.

Name	Type	Input/Output	Description
tpList	globaldefs::NamingAttributesList_T	Out	First batch of CTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining CTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the PTP does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.23 getAllDestCTPsForVCATMemberSNCProvisioning

Synopsis

```
void getAllDestCTPsForVCATMemberSNCProvisioning(
    in globaldefs::NamingAttributes_T ptpName,
    in globaldefs::NamingAttributes_T srcTPName,
    in transmissionParameters::LayerRate_T layerRate,
    in globaldefs::ConnectionDirection_T direction,
    in unsigned long how_many,
    out globaldefs:: NamingAttributesList_T tpList,
    out globaldefs:: NamingAttributesIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of CTP names that can be used as the destination of a VCAT member SNC to create on a given PTP. The NMS must specify the source CTP name (or source FTP name for VC3 port grouping circuits whose source is on E3, DS3I, or DS3IN cards), layer rate, and direction of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns a CTP name list based on current resource availability. The actual SNC creation might still fail.

Values returned by this list can be used as the zEnd in each SNCCreateData_T listed from the sncCreateDataList field of the type VCAT SNCCreateData_T, which is required as a parameter of the createVCATl method described in 3.9.26 createVCAT, page 3-156.

This interface must be used according to 2.2.7.2.4 SNC Provisioning for ONS 15454 CE-MR-10 Cards, page 2-41.

Restrict the use of this API to VCAT circuits involving CE-MR-10 cards in Manual mode.

Parameters

See the Parameters section of 3.8.22 getAllDestCTPsForSNCProvisioning, page 3-106.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the PTP does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.24 getAllSrcFTPsForSNCProvisioning

Synopsis

```
void getAllSrcFTPsForSNCProvisioning(
    in globaldefs::NamingAttributes_T meName,
    in transmissionParameters::LayerRate_T layerRate,
    in globaldefs::ConnectionDirection_T direction,
    in unsigned long how_many,
    in unsigned long members,
    out globaldefs:: NamingAttributesList_T tpList,
    out globaldefs:: NamingAttributesIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of FTP names that can be used as the source of an SNC to create on a given managed element. The NMS must specify the layer rate of the SNC that is created. This interface does not reserve resources for SNC creation; rather, this method returns the FTP name list based on current resource availability. The actual SNC creation might still fail. Values returned by this list can be used as the aEnd in the SNCCreateData_T parameter of the createAndActivateSNCFromUserLabel method described in 3.9.16 createAndActivateSNCFromUserLabel, page 3-138.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the ME that is the source of the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. All FTPs available at this layer rate are returned.
direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of FTP names to report in the first batch.
tpList	globaldefs::NamingAttributesList_T	Out	First batch of FTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining FTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.25 getAllDestFTPsForSNCProvisioning

Synopsis

```
void getAllDestFTPsForSNCProvisioning(
    in globaldefs::NamingAttributes_T meName,
    in globaldefs::NamingAttributes_T srcTPName,
    in transmissionParameters::LayerRate_T layerRate,
    in globaldefs::ConnectionDirection_T direction,
    in unsigned long how_many,
    in unsigned long no_of_member_circuits,
    out globaldefs:: NamingAttributesList_T tpList,
    out globaldefs:: NamingAttributesIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of FTP names that can be used as the destination of an SNC to create on a specific managed element. The NMS must specify the source TP (CTP or FTP) name and layer rate of the SNC to create. This interface does not reserve resources for SNC creation; rather, this method returns the FTP name list based on current resource availability. The actual SNC creation might still fail. Values returned by this list can be used as the zEnd in the SNCCreateData_T parameter of the createAndActivateSNCFromUserLabel method described in 3.9.16 createAndActivateSNCFromUserLabel, page 3-138.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the ME that is the source of the SNC to create.
srcTPName	globaldefs::NamingAttributes_T	In	Name of the TP that is selected as the source TP for the SNC to create.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate at which the SNC is created. All FTPs available at this layer rate are returned.
direction	globaldefs::ConnectionDirection_T	In	CTM supports CD_BI (bidirectional).
how_many	unsigned long	In	Maximum number of FTP names to report in the first batch.
tpList	globaldefs::NamingAttributesList_T	Out	First batch of FTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining FTP names.

For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when layerRate or direction are invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.8.26 forceNEResync

Synopsis

```
void forceNEResync(
        in globaldefs::NamingAttributes_T meName
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface forces CTM to resynchronize the inventory and alarms of the specified managed element. This interface is an asynchronous call and returns after basic validation. The managed element is then queued for resynchronization. CTM generates a state change event indicating that the emsInSyncState has changed to False. Upon successful completion of resynchronization, CTM generates a state change event indicating that the emsInSyncState attribute has changed to True. An entry is made in the CTM error log if the resynchronization fails.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the managed element to resynchronize.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or if
the managed element is already in the resynchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.27 getAllTopologicalLinksForME

Synopsis

```
void getAllTopologicalLinksForME(
        in globaldefs::NamingAttributes_T meName,
        in unsigned long how_many,
        out topologicalLink::TopologicalLinkList_T topoList,
        out topologicalLink::TopologicalLinkIterator_I topoIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of topological links that originate or terminate on the specified managed element.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the managed element.
how_many	unsigned long	In	Maximum number of topological links to return in the first batch.
topoList	topologicalLink::TopologicalLinkList_T	Out	First batch of topological links.
tpIt	topologicalLink::TopologicalLinkIterator_I	Out	Iterator to retrieve the remaining topological links.

For details about the TopologicalLink_T structure, see 3.3.5 getAllTopLevelTopologicalLinks, page 3-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.8.28 getAllRolls

Synopsis

```
void getAllRolls(
        in globaldefs::NamingAttributes_T meName,
        in unsigned long how_many,
```

```
    out multiLayerSubnetwork::RollList_T rollList,
    out multiLayerSubnetwork::RollIterator_I rollIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of rolls within the managed element whose name is passed as a parameter.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of MEs with rolls that are being queried.
how_many	unsigned long	In	Maximum number of rolls to report in the first batch.
rollList	multiLayerSubnetwork::RollList_T	Out	First batch of rolls.
rollIt	multiLayerSubnetwork::RollIterator_I	Out	Iterator to retrieve the remaining rolls.

CTM supports Roll_T as follows:

- globaldefs::NamingAttributes_T rollName

  Supported. It contains three tuples: the first tuple contains the EMS name, the second tuple contains theManagedElement name, and the third tuple contains the nativeEMSName of this roll.

- string::userLabel

  Not supported.

- string::nativeEMSName

  Supported. Each roll has a unique name assigned by the Cisco MSM.

- string::owner

  Not supported.

- multiLayerSubnetwork::RollMode_T rollMode

  Supported. Possible values are AUTO or MANUAL.

- globaldefs::NamingAttributes_T rollFromSNCName

  Supported. The name of the rollFrom SNC.

- globaldefs::NamingAttributes_T rollToSNCName

  Supported. The name of the rollTo SNC.

- globaldefs::NamingAttributes_T rollFromCTPName

  Supported. The name of the CTP that is rolled over.

- globaldefs::NamingAttributes_T rollToCTPName

  Supported. The CTP to which the rollFromCTPName CTP is rolled over.

- multiLayerSubnetwork::RollState_T rollState

  Supported. The following values are supported:

  - ROLL_PENDING
  - ROLL_COMPLETED
  - ROLL_CANCELLED
  - ROLL_INCOMPLETE

- boolean rollValidSignal

  Supported. Values are TRUE or FALSE.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or in
config synchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.29 getAllRollNames

Synopsis

```
void getAllRollNames(
        in globaldefs::NamingAttributes_T meName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of roll names within the managed element whose name is passed as a parameter.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the managed element with rolls that are being queried.
how_many	unsigned long	In	Maximum number of rolls to report in the first batch.
nameList	globaldefs::NamingAttributesList	Out	First batch of roll names.
nameIt	globaldefs::namingAttributesIterator_I	Out	Iterator to retrieve the remaining roll names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or in
config synchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.
```

```
EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.30 getRoll

Synopsis

```
void getRoll(
        in globaldefs::NamingAttributes_T rollName,
        out multiLayerSubnetwork::Roll_T roll,
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the Roll_T structure for a specific roll name.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll.
Roll	multiLayerSubnetwork::Roll_T	Out	Roll object.

CTM support for Roll_T is described in 3.8.28 getAllRolls, page 3-111.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or in
config synchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the roll does not exist.

EXCPT_INVALID_INPUT - Raised if the roll name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.31 getRollToPTPs

Synopsis

```
void getRollToPTPs (
       in globaldefs::NamingAttributes_T meName,
       in globaldefs::NamingAttributesList_T sncNames,
       in unsigned long how_many,
       out globaldefs::NamingAttributesList_T tpList,
```

```
        out globaldefs::NamingAttributesIterator_I tpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of PTPs on a managed element that can be used for provisioning a roll on a specific SNC. The returned PTPs can be used to contain the rollToCTP. The SNC must be associated with the managed element on which the interface is invoked.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes_T	In	Name of the managed element.
sncNames	globaldefs::NamingAttributesList_T	In	Name of the SNC to roll over. If more than one SNC is specified, the first SNC becomes the RollFrom SNC and the second SNC becomes the RollToSNC.
how_many	Unsigned	In	Maximum number of PTPs to report in the first batch.
tpList	globaldefs::namingAttributesList_T	Out	First batch of PTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining PTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or in
config synchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.32 getRollToCTPs

Synopsis

```
void getRollToCTPs(
       in globaldefs::NamingAttributes_T ptpName,
       in globaldefs::NamingAttributesList_T sncNames,
       in unsigned long how_many,
       out globaldefs::NamingAttributesList_T tpList,
       out globaldefs::NamingAttributesIterator_I tpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of CTPs contained in the PTP. This list can be used for roll provisioning as the rollToCTP for a given SNC. The list of PTPs can be obtained by using the operation defined in 3.8.31 getRollToPTPs, page 3-114.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP for which the list of CTPs is obtained for roll provisioning.
how_many	Unsigned	In	Maximum number of PTPs to report in the first batch.
sncNames	globaldefs::NamingAttributesList_T	In	Name of the SNC to roll over. If more than one SNC is specified, the first SNC becomes the RollFrom SNC and the second SNC becomes the RollToSNC.
tpList	globaldefs::namingAttributesList_T	Out	First batch of CTP names.
tpIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve remaining CTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or in
config synchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.8.33 getRollFromCTPs

Synopsis

```
void getRollFromCTPs(
       in globaldefs::NamingAttributes_T meName,
       in globaldefs::NamingAttributesList_T sncNames,
       in globaldefs::NamingAttributes_T fixCTPName,
       in unsigned long how_many,
       out globaldefs::NamingAttributesList_T tpList,
       out globaldefs::NamingAttributesIterator_I tpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of rollFromCTPNames after selecting the fixed point of the roll. This is needed for path-protection SNCs and SNCs that have a secondary source, because multiple rollFromCTP values are possible.

This interface also returns a list of CTPs that can be used to populate the rollFromCTP data structure in the RollCreateData_T data structure.

Parameters

Name	Type	Input/Output	Description
sncNames	globaldefs::NamingAttributes List_T	In	Name of the SNC to roll over. If more than one SNC is specified, the first SNC becomes the RollFrom SNC and the second SNC becomes the RollToSNC.
fixCTPName	globaldefs::NamingAttributes _T	In	Fixed point on the roll.
how_many	Unsigned	In	Maximum number of CTPs to report in the first batch.
tpList	globaldefs::namingAttributes List_T	Out	First batch of CTP names.
tpIt	globaldefs::NamingAttributesI terator_I	Out	Iterator to retrieve the remaining CTP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the cancellation of the rolling operation.

EXCPT_UNABLE_TO_COMPLY - Raised when the roll cannot be created.
```

Compliance

CTM-defined.

3.8.34 setPortName

Synopsis

```
void setPortName(
        in globaldef::NamningAttributes_T ptpName,
        in string portName,
        out string newPortName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to set a name for the selected port. The PTP specified as input identifies the port. The port name is set to the value passed in through the portName parameter.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributesList_T	In	Name of the PTP whose name will be set.
portName	String	In	Port name value to set for the specified PTP.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when the ptpName does not refer to a valid PTP name or when
it contains undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when ptpName references an object that does not exist.
```

Compliance

CTM-defined.

3.8.35 getPortName

Synopsis

```
void getPortName (
        in globaldefs:: NamingAttributes_T ptpName,
        out string portName )
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to get a name for the selected port. The PTP specified as input identifies the port. The resulting port name is used as the argument parameter.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributesList_T	In	Name of the PTP whose name will be set.
portName	String	Out	Port name specified for the input PTP.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when the ptpName does not refer to a valid PTP name or when
it contains undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when ptpName references an object that does not exist.
```

Compliance

CTM-defined.

3.8.36 getAllClearedAlarms

Synopsis

```
void getAllClearedAlarms(
        in globaldefs::NamingAttributes_T meName,
        in notifications::ProbablyCauseList_T excludeProbCauseList,
        in notifications::PerceivedSeverityList_T excludeSeverityList,
```

```
    in unsigned long how_many,
    out notifications::EventList_T eventList,
    out notifications::EventIterator_I eventIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request all of the cleared and acknowledged alarms for the specified ME. Alarms that are not reported by the ME to the EMS are not reported by this operation. Some alarms may be filtered out by specifying the probable cause of the severities.

The result of this operation is independent of the filtering set up by the NMS for the notification service.

ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH

The ONS 15310 CL, ONS 15310 MA SDH, ONS 15310 MA SONET, ONS 15327, ONS 15454 SONET, ONS 15454 SDH, ONS 15600 SONET, and ONS 15600 SDH do not send a “clear” for TCA events. TCAs are not reported in the eventList returned by this method.

ONS 15530 and ONS 15540

TCAs are reported in the eventList returned by this method. The ONS 15530 and ONS 15540 send a “clear” for TCAs.

Parameters

Name	Type	Input/Output	Description
meName	globaldefs::NamingAttributes List_T	In	Name of the ME for which to retrieve alarms.
excludeProbCause List	notifications::ProbableCause List_T	In	List of probable causes to exclude (for which events are not reported).
exludeSeverityList	notifications::PerceivedSeveri tyList_T	In	List of severities to exclude from the output event list.
how_many	Unsigned long	In	Maximum number of alarms to report in the first batch.
eventList	notifications::EventList	Out	First batch of alarms.
eventIt	notifications::EventIterator_I	Out	Iterator to retrieve the remaining alarms.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when meName does not reference a managed element object or
excludeProbCauseList contains undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an ME object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.8.37 getAllCrossConnections

Synopsis

```
void getAllCrossConnections(in globaldefs::NamingAttributes_T managedElementName,
      in transmissionParameters::LayerRateList_T connectionRateList,
      in unsigned long how_many,
      out subnetworkConnection::CrossConnectList_T ccList,
      out subnetworkConnection::CCIterator_I ccIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of cross-connections on the managed element.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the managed element.
connectionRateList	transmissionParameters::LayerRateList_T	In	Layer rate list.
how_many	unsigned long	In	Maximum number of CrossConnect_T to report in the first batch.
ccList	subnetworkConnection::CrossConnectList_T	Out	First batch of cross-connects.
ccIt	subnetworkConnection::CCIterator_I	Out	Iterator to retrieve the remaining cross-connects.

A cross-connect represents a connection within a single managed element. This structure is used primarily in the specification of routes.

CTM support for the CrossConnection_T structure is as follows:

- boolean active:
  - Always true. CTM only returns cross-connections for SNCs that are in SNCState_T.SNCS_ACTIVE or SNCState_T.SNCS_PARTIAL. This attribute indicates whether the cross-connect is active in the ME.
- globaldefs::ConnectionDirection_T direction:
  - Direction of the cross-connection. CTM returns CD_UNI or CD_BI.
- SNCType_T ccType:
  - CTM always returns ST_SIMPLE.
- globaldefs::NamingAttributesList_T aEndNameList:
  - Names of TPs at the aEnd of the cross-connection.
- globaldefs::NamingAttributesList_T zEndNameList:
  - Names of TPs at the zEnd of the cross-connection.
- globaldefs::NVSList_T additionalInfo:
  - This attribute allows the EMS to communicate to the NMS additional information that is not explicitly modeled. This attribute is read-only. CTM does not return any additional information.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised when managedElementName references an ME
object that does not exist.

EXCPT_INVALID_INPUT - Raised if managedElementName does not reference a
managed element.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

TMF-defined.

3.8.38 getAllSoftwareDownloadJobs

Synopsis

```
void getAllSoftwareDownloadJobs(
         in unsigned long how_many,
         out JobMonitorList_T jobList)
         raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the status of all scheduled and completed software download jobs.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Number of download jobs to retrieve.
jobList	JobMonitorList_T	Out	List of jobs scheduled (see the definition that follows this table).

```
typedef sequence<JobMonitorData_T> JobMonitorList_T;
struct JobMonitorData_T{
        unsigned long jobId;
        unsigned long taskId;
        string taskType;
        string taskOwner;
        string neId;
        string taskStatus;
        string creationTime;
        string scheduledTime;
        string startTime;
        string endTime;
        string displayInformation;
        string userComments;
        string additionalComments;
};
```

The following table lists descriptions of each individual parameter.

Name	Type	Description
jobId	string	The job ID.
taskId	string	The task ID.
taskType	string	The kind of task.
taskOwner	string	The task owner.
neId	string	The managed element name.
taskStatus	string	The status of the task. Valid values are: • 1: QUEUED • 2: RUNNING • 3: CANCELLED • 4: FAILED • 5: SUCCEEDED • 6: WAITING
creationTime	string	The time when the task was created.
scheduledTime	string	The scheduled time for the task.
startTime	string	The actual start time for the task.
endTime	string	The actual time of completion.
displayInformation	string	Display information.
userComments	string	User comments.
additionalComments	string	Additional comments.

Throws

```
globaldefs::ProcessingFailureException
```

Compliance

CTM-defined.

3.8.39 setMEAdminState

Synopsis

```
void setMEAdminState  (in globaldefs::NamingAttributes_T meName,
    in AdminState_T adminState)
raises         (globaldefs::ProcessingFailureException)
```

Description

This interface is used to force the EMS to change the administrative state of the managed element to the specified state.

Parameters

Name	Type	Input/Output	Description
meName	NamingAttributes_T	In	Name of the managed element to resynchronize. It must contain a name/value pair, where the name is *ManagedElement* and the value is *ME Name*.
adminState	AdminState_T	In	Admin state to which the managed element name changes.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_UNABLE_TO_COMPLY - Raised if the managed element is in the initial poll state or if
the managed element is already in the resynchronization state.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised if the managed element name is invalid.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.
```

The following enum is used to define the various NE admin states:

```
enum  AdminState_T {
IN_SERVICE,
OUT_OF_SERVICE
 OUT_OF_SERVICE_BY_MAINTENANCE
}
```

Compliance

CTM-defined.

3.9 multiLayerSubnetwork::MultiLayerSubnetworkMgr_I

The multiLayerSubnetworkMgr_I operation is used to gain access to subnetworks and their respective operations. A handle to an instance of this interface is gained through the getManager operation in emsSession. This interface is inherited from common::Common_I.

```
interface ManagedElementManager_I:common::Common_I
```

multiLayerSubnetworkMgr_I also implements the getCapabilities method. Operations described in following subsections are returned to the NMS when it invokes getCapabilities.

3.9.1 getAllEdgePoints

Synopsis

```
void getAllEdgePoints(
        in globaldefs::NamingAttributes_T subnetName,
        in transmissionParameters::LayerRateList_T tpLayerRateList,
        in transmissionParameters::LayerRateList_T connectionLayerRateList,
```

```
    in unsigned long how_many,
    out terminationPoint::TerminationPointList_T tpList,
    out terminationPoint::TerminationPointIterator_I tpIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a list of edge termination points for the specified subnetwork at one or more of the NMS-specified layers that are capable of containing CTPs that can be connected at one or more of the NMS-specified connection layer rates.

CTM reports the PTPs that are not part of the topological link.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
tpLayerList	transmissionParameters::LayerRate List_T	In	List of TP layer rates for which edge points are retrieved. To be reported, an edge point must contain at least one of the layer rates specified. If the list is empty, edge points of all rates are returned.
connectionLayerRate List	transmissionParameters::LayerRate List_T	In	CTM does not use this parameter.
how_many	unsigned long	In	Maximum number of edge points to return in the first batch.
tpList	terminationPoint::TerminationPoint List_T	Out	First batch of edge points.
tpIt	terminationPoint::TerminationPoint Iterator_I	Out	Iterator to retrieve the remaining edge points.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object, or tpLayerRateList or connectionLayerRateList contain undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.2 getAllEdgePointNames

Synopsis

```
getAllEdgePointNames
```

```
void getAllEdgePointNames(in globaldefs::NamingAttributes_T subnetName,
    in transmissionParameters::LayerRateList_T layerRateList,
    in transmissionParameters::LayerRateList_T connectionLayerRateList,
    in unsigned long how_many,
    out globaldefs::NamingAttributesList_T nameList,
    out globaldefs::NamingAttributesIterator_I nameIt)
  raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a list of edge termination points for the specified subnetwork at one or more of the NMS-specified layers that are capable of containing CTPs that can be connected at one or more of the NMS-specified connection layer rates.

CTM reports the PTPs that are not part of the topological link.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
tpLayerList	transmissionParameters::LayerRateList_T	In	List of TP layer rates for which the edge points are fetched. To be reported, an edge point must contain at least one of the layer rates specified. If the list is empty, edge points of all rates are returned.
connectionLayerRateList	transmissionParameters::LayerRateList_T	In	CTM does not use this parameter.
how_many	unsigned long	In	Maximum number of edge points to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of edge point names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining edge point names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object or tpLayerRateList or connectionLayerRateList contain undefined values.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.3 getAllManagedElement

Synopsis

```
void getAllManagedElementNames(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a list of managed elements that are associated with the specified subnetwork.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
how_many	unsigned long	In	Maximum number of MEs to report in the first batch.
meList	managedElement::ManagedElementList_T	Out	First batch of ME names.
MeIt	managedelement::ManagedElementIterator_I	Out	Iterator to retrieve the remaining ME names.

For details about the ManagedElement_T structure, see 3.8.3 getAllManagedElements, page 3-85.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.4 getAllManagedElementNames

Synopsis

```
void getAllManagedElementNames(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.9.3 getAllManagedElement, page 3-126, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
how_many	unsigned long	In	Maximum number of MEs to report in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of ME names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining ME names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.5 getAllSubnetworkConnections

Synopsis

```
void getAllSubnetworkConnections(
        in globaldefs::NamingAttributes_T subnetName,
        in transmissionParameters::LayerRateList_T connectionRateList,
        in unsigned long how_many,
        out subnetworkConnection::SubnetworkConnectionList_T sncList,
        out subnetworkConnection::SNCIterator_I sncIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a list of SNCs for the specified subnetwork at the specified connection rates.

- For the ONS 15310 MA SONET, this method does not report STS-1 SNC on DS1 ports of DS1_xx cards.
- For the ONS 15310 MA SDH, this method does not report VC4 SNC on E1 ports of E1_63_E3_3 cards.
- For the ONS 15327, this method does not report STS-1 SNC on XTC using DS1 ports.
- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report SNCs with duplicate names or VCAT SNCs.

If the CTM subnetwork contains multiple circuits with the same name, the getAllSubnetworkConnections call returns the circuit name and details for only one of the circuits. It is not possible to predict which one of the multiple circuits will have its name and details returned.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	in	Name of the subnetwork.
connectionLayerRateList	transmissionParameters::LayerRateList_T	in	List of rates of the SNCs to report. If an empty list is specified, all SNCs of all rates are reported.
how_many	unsigned long	in	Maximum number of SNCs to report in the first batch.
sncList	subnetworkConnection::SubnetworkConnectionList_T	out	First batch of SNCs.
sncIt	subnetworkConnection::SNCIterator_I	out	Iterator to retrieve the remaining SNCs.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.6 getAllSubnetworkConnectionsWithTP

Synopsis

```
void getAllSubnetworkConnectionsWithTP(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T connectionRateList,
        in unsigned long how_many,
        out subnetworkConnection::SubnetworkConnectionList_T sncList,
        out subnetworkConnection::SNCIterator_I sncIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a list of the SNCs using the specified TP at the specified connection rates. A TP might be a PTP that requires a full list of SNCs using any of its contained CTPs. A TP might be a CTP or FTP that only requires the SNCs using that CTP or FTP. Both end CTPs and intermediate CTPs at the same layer are considered. In all cases, only SNCs that also match the specified connection rate filter are returned.

- For the ONS 15310 MA SONET, this method does not report STS-1 SNC on DS1 ports of DS1_xx cards.
- For the ONS 15310 MA SDH, this method does not report VC4 SNC on E1 ports of E1_63_E3_3 cards.
- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report SNCs with duplicate names.

Note For ONS 15600 SONET and ONS 15600 SDH NEs, if bridge-and-roll is applied, one CTP may be associated with two SNCs during a transient phase. If CTM invokes this method during this phase, it reports both SNCs.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	in	Termination point for which to report SNCs.
connectionLayerRateList	transmissionParameters::LayerRateList_T	in	List of rates of the SNCs to report. If an empty list is specified, all SNCs of all rates are reported.
how_many	unsigned long	in	Maximum number of SNCs to report in the first batch.
sncList	subnetworkConnection::SubnetworkConnectionList_T	out	First batch of SNCs.
sncIt	subnetworkConnection::SNCIterator_I	out	Iterator to retrieve the remaining SNCs.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if tpName does not reference a termination point object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.7 getAllSubnetworkConnectionNames

Synopsis

```
void getAllSubnetworkConnectionNames(
        in globaldefs::NamingAttributes_T subnetName,
        in transmissionParameters::LayerRateList_T connectionRateList,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.9.5 getAllSubnetworkConnections, page 3-127, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
connectionLayerRateList	transmissionParameters::LayerRateList_T	In	List of rates of the SNCs to report. If an empty list is specified, all SNCs of all rates are reported.
how_many	unsigned long	In	Maximum number of SNCs to report in the first batch.
nameList	globaldefs::NamingAttributesList	Out	First batch of SNC names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining SNC names.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.8 getAllSubnetworkConnectionNamesWithTP

Synopsis

```
void getAllSubnetworkConnectionNamesWithTP(
        in globaldefs::NamingAttributes_T tpName,
        in transmissionParameters::LayerRateList_T connectionRateList,
```

```
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.9.6 getAllSubnetworkConnectionsWithTP, page 3-128, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	in	Termination point for which to report SNCs.
connectionLayerRateList	transmissionParameters::LayerRateList_T	in	List of rates of the SNCs to report. If an empty list is specified, all SNCs of all rates are reported.
how_many	unsigned long	in	Maximum number of SNCs to report in the first batch.
nameList	globaldefs::NamingAttributesList	out	First batch of SNC names.
nameIt	globaldefs::NamingAttributesIterator_I	out	Iterator to retrieve the remaining SNC names.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if tpName does not reference a terminationPoint object.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.9 getAllTopologicalLinks

Synopsis

```
void getAllTopologicalLinks(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out topologicalLink::TopologicalLinkList_T topoList,
        out topologicalLink::TopologicalLinkIterator_I topoIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of topological links that exist within the subnetwork whose name is passed as a parameter.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
how_many	unsigned long	In	Maximum number of topological links to return in the first batch.
topoList	topologicalLink::TopologicalLinkList_T	Out	First batch of topological links.
topoIt	topologicalLink::TopologicalLinkIterator_I	Out	Iterator to retrieve the remaining topological links.

For details about the TopologicalLink_T structure, see 3.3.5 getAllTopLevelTopologicalLinks, page 3-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.10 getAllTopologicalLinkNames

Synopsis

```
void getAllTopologicalLinkNames(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface has the same behavior as the operation described in 3.9.9 getAllTopologicalLinks, page 3-131, but instead of returning entire object structures, this operation returns their names.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
how_many	unsigned long	In	Maximum number of topological links to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of topological link names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining topological link names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.9.11 getMultiLayerSubnetwork

Synopsis

```
void getMultiLayerSubnetwork(
        in globaldefs::NamingAttributes_T subnetName,
        out MultiLayerSubnetwork_T subnetwork)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a subnetwork given a subnetwork name.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
subnetwork	MultiLayerSubnetwork_T	Out	Subnetwork structure returned.

For details about the MultiLayerSubnetwork_T structure, see 3.3.3 getAllTopLevelSubnetworks, page 3-21.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references a multiLayerSubnetwork object
that does not exist.
```

Compliance

TMF-defined.

3.9.12 getCTPsAndTopologicalLinks

Synopsis

```
void getCTPsAndTopologicalLinks(
        in globaldefs::NamingAttributes_T sncName,
        out terminationPoint::TerminationPointList_T ctpList,
        out topologicalLink::TopologicalLinkList_T topologicalLinkList)
     raises(globaldefs::ProcessingFailureException);
```

Description

The NMS provides the SNC name, and CTM returns ordered lists: one list for all CTPs/FTPs and another for all topological links. CTM reports topological links and CTPs/FTPs of the working path for a 1+1, 2-fiber BLSR, or 4-fiber BLSR protected SNC. CTM reports topological links and CTPs/FTPs of both the working and protect paths for a path-protection SNC. An SNC with the source or destination on Ethernet equipment is also supported.

- For the ONS 15310, this method does not report STS-1 SNC on CTX using DS1 ports.
- For the ONS 15327, this method does not report STS-1 SNC on XTC using DS1 ports.
- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report VCAT SNCs.

The lists are unordered for circuits for which:

- The circuit state is not active.
- The circuit has DRI protection.

A processing failure exception is returned with an EXCPT_UNABLE_TO_COMPLY exception in all of the preceding conditions.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the subnetwork connection.
ctpList	terminationPoint::TerminationPointList_T	Out	Ordered list of CTP/FTPs for this SNC.
topoList	topologicalLink::TopologicalLinkList_T	Out	Ordered topological links through which this SNC traverses.

For details about the TerminationPoint_T structure, see 3.6.3 getAllSupportedPTPs, page 3-63.

For details about the TopologicalLink_T structure, see 3.3.5 getAllTopLevelTopologicalLinks, page 3-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_UNABLE_TO_COMPLY - Raised when invoked on SNC on Ethernet equipment or STS-1 SNC on
DS1 equipment.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an SNC object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.9.13 getSNC

Synopsis

```
void getSNC(
        in globaldefs::NamingAttributes_T sncName,
        out subnetworkConnection::SubnetworkConnection_T snc)
        raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the SNC structure for the SNC whose name is supplied as a parameter.

- For the ONS 15310 CL, this method does not report STS-1 SNC on CTX using DS1 ports.
- For the ONS 15310 MA SONET, this method does not report STS-1 SNC on DS1 ports of DS1_xx cards.
- For the ONS 15310 MA SDH, this method does not report VC4 SNC on E1 ports of E1_63_E3_3 cards.
- For the ONS 15327, this method does not report STS-1 SNC on XTC using DS1 ports.
- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report VCAT SNCs.

A processing failure exception is returned with an EXCPT_UNABLE_TO_COMPLY exception in all of the preceding conditions.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the subnetwork connection.
snc	subnetworkConnection::SubnetworkConnection_T	Out	SNC structure retrieved.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not reference an SNC object or when sncName
is not unique.

EXCPT_UNABLE_TO_COMPLY - Raised when invoked on SNC on Ethernet equipment or STS-1 SNC on
DS1 equipment.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an SNC object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.9.14 getSNCsByUserLabel

Synopsis

```
void getSNC(
        in string userLabel,
        out subnetworkConnection::SubnetworkConnection_T snc)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the SNC structure for the SNC whose name is supplied as a parameter.

- For the ONS 15310 CL, this method does not report STS-1 SNC on CTX using DS1 ports.
- For the ONS 15310 MA SONET, this method does not report STS-1 SNC on DS1 ports of DS1_xx cards.
- For the ONS 15310 MA SDH, this method does not report VC4 SNC on E1 ports of E1_63_E3_3 cards.
- For the ONS 15327, this method does not report STS-1 SNC on XTC using DS1 ports.
- For the ONS 15454 SONET, this method does not report STS-1 SNC on DS1 equipment.
- For the ONS 15454 SDH, this method does not report VC-4 SNC on E1 or E1_42 equipment.

This interface does not report VCAT SNCs.

A processing failure exception is returned with an EXCPT_UNABLE_TO_COMPLY exception in all of the preceding conditions.

Parameters

Name	Type	Input/Output	Description
userLabel	String	In	Name of the subnetwork connection.
snc	subnetworkConnection::SubnetworkConnection_T	Out	SNC structure retrieved.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not reference an SNC object or when sncName
is not unique.

EXCPT_UNABLE_TO_COMPLY - Raised when invoked on SNC on Ethernet equipment or STS-1 SNC on
DS1 equipment.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an SNC object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.9.15 getTopologicalLink

Synopsis

```
void getTopologicalLink(
        in globaldefs::NamingAttributes_T topoLinkName,
        out topologicalLink::TopologicalLink_T topoLink)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a topological link. CTM reports all autodiscovered and manual topological links.

Parameters

Name	Type	Input/Output	Description
topoLinkName	Globaldefs::NamingAttributes_T	In	Name of the subnetwork topological link to retrieve.
topoLink	topologicalLink::TopologicalLink_T	Out	Subnetwork topological link returned.

For details about the TopologicalLink_T structure, see 3.3.5 getAllTopLevelTopologicalLinks, page 3-23.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when topoLinkName does not reference a subnetwork topological
link object.

EXCPT_ENTITY_NOT_FOUND - Raised when topoLinkName references a subnetwork topological link
object that does not exist.
```

Compliance

TMF-defined.

3.9.16 createAndActivateSNCFromUserLabel

Synopsis

```
void createAndActivateSNCFromUserLabel(
        in subnetworkConnection::SNCCreateData_T createData,
        in subnetworkConnection::GradesOfImpact_T tolerableImpact,
        in EMSFreedomLevel_T emsFreedomLevel,
        inout subnetworkConnection::TPDataList_T tpsToModify,
        out subnetworkConnection::SubnetworkConnection_T theSNC,
        out string errorReason)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a way to create and activate a subnetwork connection in one command. It is CTM-specific and not TMF-defined. According to the TMF, the SNC name is assigned by the EMS when it creates the SNC. However, if the NMS is to control the SNC name, it must use this method. The NMS provides the name of the SNC in the userLabel field of the SNCCreateData input parameter. CTM uses the userLabel value and assign it to the name of the new SNC. CTM also verifies the uniqueness of the name. The operation fails if an SNC with the same name as the userLabel exists. This is an asynchronous operation. Successful completion of this operation only guarantees delivery of the creation request to CTM. Operation on CTPs of this SNC might still fail.

Optionally, you can also provide a description for the SNC by entering it in the additionalCreationInfo field.

CTM cleans up resources if SNC creation fails. For example, if CTM fails to create the SNC on the destination node, CTM also cleans up resources on the source node.

If the OC-n port is configured for SDH, CTM can create the SNC. CTM supports only those layer rates supported by the NE. However, IPPM and J1 path tracing are not supported for CTPs of this SNC.

FTPs can be used as A ends and Z ends of the SNCs. A typical use of FTP is to create an SNC on Ethernet equipment, create a tunnel SNC, or create a VAP/LAP circuit. See 2.2.7 Provisioning Subnetwork Connections, page 2-39 for more information.

Creating SNCs between ONS 15327 nodes or ONS 15310, and between ONS 15600 nodes is identical to creating SNCs between ONS 15454 nodes. It is also possible to create SNCs among ONS 15327, ONS 15454, and ONS 15600 nodes.

CTM supports protection channel access (PCA). PCA SNCs are unprotected. For details about PCA, see the NE documentation.

You can provide a description for the SNC. Enter the description in the additionalCreationInfo field.

To create a dual-ring interconnect SNC, set the staticProtectionLevel value in createData to HIGHLY_PROTECTED. Use the CTC or CTM client to modify the Hold-off Timer attribute for this SNC.

The following types of circuit creation are not supported:

- Monitor circuits
- STS-1 on DS1 cards and VC-4 on E1 and E1_42 cards
- DRI circuits with constraint routing

In addition to specifying SNC creation parameters using the SNCCreateData_T structure, the NMS can provide a partial or complete list of MEs or both end PTPs of a topological link to include or exclude in the SNC route during SNC creation. The NMS can specify only one type of object name list during the SNC creation. CTM returns a processing failure exception if the NMS-specified list is not homogeneous. CTM returns a processing failure exception with EXCPT_ENTITY_NOT_FOUND as the error reason

if any one of the specified MEs or PTPs does not exist. If the NMS specifies a PTP list, CTM verifies whether PTPs are specified in pairs and each PTP of a PTP pair is associated with each end PTP of the same topological link. CTM uses the first PTP of a PTP pair as the aEnd of the topological link, and uses the second PTP of the same PTP pair as the zEnd of the topological link. CTM attempts to route the SNC from the aEnd PTP of a PTP pair to the zEnd PTP of the same pair on a topological link. CTM returns a processing failure exception if the SNC cannot be routed from the specified aEnd PTP to the zEnd PTP of a topological link. The NMS then must interchange the aEnd PTP and zEnd PTP specified, or provide a different PTP pair, or delete the PTP pair from the include or exclude list. CTM returns a processing failure exception if at least one PTP is not associated with a topological link. The NMS must ensure that the sequence of the object name list conforms to the sequence of the objects that exist in the network. CTM returns a processing failure exception if the sequence is nonconforming.

In addition to specifying SNC creation parameters using the SNCCreateData_T structure, the NMS can provide a complete list of CTP pairs representing the complete route of the SNC. Each CTP pair represents the two endpoints of one span in the SNC route. A span represents a partial route of an SNC with the endpoints specified on the endpoints of a topological link. CTM processes the input list in CTP pairs. CTM assumes each pair represents the two endpoints of a span, and the CTP native names are the same for the two endpoints (the STS numbers are the same for each CTP pair). The NMS must ensure that the sequence of spans represented by the CTP pairs conforms to the sequence existing in the network. CTM returns a processing failure exception if the sequence is nonconforming.

If you create an SNC with at least one endpoint on ML-series equipment, CTM identifies the SNC for a new Layer 2 point-to-point topology. CTM creates a new Layer 2 point-to-point topology and an object creation event is generated to notify for the creation of a new L2 point-to-point topology in CTM.

CTM provides support for the creation of open-ended path-protection circuits. There are two types of SNCs that can be created for open-ended path-protection circuits.

For one source point and two destination endpoints, the NMS must specify the source point in SNCCreateData_T.aEnd, and the two destination endpoints in SNCCreateData_T.zEnd.

For two source points and one destination endpoint, the NMS must specify two source points in SNCCreateData_T.aEnd and one destination endpoint in SNCCreateData_T.zEnd.

CTM supports optical channel SNC creation on WDM cards through this method. See 2.2.7.4 SNC Provisioning on WDM Equipment, page 2-43 for detailed information. The sncSize field in additionalCreationInfo is used to specify the size of the optical channel SNC.

CTM supports OCH trail SNC creation on DWDM transponder/muxponder cards or ITU-T line cards through a DWDM network. See 2.2.7.4 SNC Provisioning on WDM Equipment, page 2-43.

For the ONS 15454 SONET, this operation enables you to create an end-to-end STS-1 circuit using the portless transmux interfaces of the transmux card as an intermediate connection point. Information about the high-density portless transmux card (DS3XM-12) can be provided using the usePortlessTransmux, portlessTransmuxMeName, portlessTransmuxSlot, and isSourceDS3MappedSTS attributes in the additionalCreationInfo field.

Parameters

Name	Type	Input/Output	Description
createData	subnetworkConnection::SNCCreateData_T	In	Structure describing the SNC to create and activate.
tolerableImpact	subnetworkConnection::GradesOfImpact_T	In	Maximum tolerable impact allowed.
emsFreedomLevel	EMSFreedomLevel_T	In	Maximum level of freedom allowed to the EMS to perform the creation and activation.

Name	Type	Input/Output	Description
tpsToModify	subnetworkConnection::TPDataList_T	In/out	CTM ignores this parameter.
theSNC	subnetworkConnection::SubnetworkConnection_T	Out	Resulting SNC, with the sncState and name set. The name is set to the user label supplied in the createData structure.
errorReason	String	Out	Specifies the creation and/or activation error(s), if any.

CTM does not support locking.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

CTM support for the SNCCreateData_T structure is as follows:

- string userLabel

  Supported. The value in this field is assigned to the name of the newly created SNC.

- boolean forceUniqueness

  CTM guarantees the uniqueness of SNCs if they are all created by CTM. CTM cannot guarantee the uniqueness if any other tool has been used instead.

- string owner

  Not supported.

- globaldefs::ConnectionDirection_T direction

  CTM supports CD_UNI (unidirectional), CD_BI (bidirectional), CD_UNI_EW (unidirectional east-to-west), CD_UNI_WE (unidirectional west-to-east), CD_BI_EW (bidirectional east-to-west), and CD_BI_WE (bidirectional west-to-east).

- StaticProtectionLevel_T staticProtectionLevel

  CTM supports UNPROTECTED, FULLY_PROTECTED, and HIGHLY_PROTECTED. CTM does not support any other value.

- ProtectionEffort_T protectionEffort

  CTM supports all values defined by the TMF.

  If the NMS requests an unprotected SNC on a BLSR ring, CTM creates the SNC. The same is true for an unprotected SNC with 1+1 protection.

  If the NMS specifies EFFORT_SAME_OR_WORSE or EFFORT_WHATEVER, CTM expects the value of staticProtectionLevel to be UNPROTECTED. These values indicate that CTM must create the PCA.

- Reroute_T rerouteAllowed

  Not supported.

- NetworkRouted_T networkRouted

  Not supported.

- SNCType_T sncType;

  Not supported.

- transmissionParameters::LayerRate_T layerRate

  CTM supports the following layer rates:

Layer Rate Category	Layer Rates	SNC Sizes	Comments
Video	LR_D1_Video LR_HDTV LR_DV_6000 LR_DVBASI	27=SD1/D1 26=HDTV 28=DV 6000 31=Video /DVB ASI	For WDM OCHCC
Data/Storage	LR_ESCON LR_ETR LR_ISC3_PEER_1G LR_ISC3_PEER_2G LR_ISC1 LR_IB LR_SYSPLEX_ISC_COMP LR_SYSPLEX_ISC_PEER	20=ESCON 21=ETR-CLO 22=ISC PEER” 24=ISC3 PEER 2Gb/s 23=ISC3 PEER 1Gb/s 25=ISC COMPAT 155=ISC1 30=data Storage /Video - ISC Channel 32=IB	For WDM OCHCC
STS/VC	LR_STS1_and_AU3_High_Order_VC3 LR_STS3c_and_AU4_VC4 LR_STS12c_and_VC4_4c LR_STS48c_and_VC4_16c LR_STS192c_and_VC4_64c LR_STS6c_and_VC4_2c LR_STS9c_and_VC4_3c LR_STS24c_and_VC4_8c LR_STS18c_and_VC4_6c LR_STS36c_and_VC4_12c LR_VT1_5_and_TU11_VC11 LR_Low_Order_TU3_VC3 LR_VT2_and_TU12_VC12		Only on ONS15454 SDH
FICON	LR_FICON_1G LR_FICON_2G LR_FICON_4G LR_FC_100_1063M LR_FC_200_2125M	19=1Gb/s FiCon 18=2Gb/s FiCon 17=4Gb/s FiCon	For WDM OCHCC

Layer Rate Category	Layer Rates	SNC Sizes	Comments
FC	LR_FC_100_1063M LR_FC_200_2125M LR_FC_400_4250M LR_FC_800_8500M LR_FC_1000_10518M	16=1Gb/s FC 15=2Gb/s FC 14=4Gb/s FC 86= 8Gb/s FC 13=10Gb/s FC	For WDM OCHCC
SONET/SDH	LR_DSR_OC3_STM1 LR_DSR_OC12_STM4 LR_DSR_OC48_STM16 LR_DSR_OC192_STM64 LR_DSR_OC768_and_STM256	7=OC 192 8=OC 48 9=OC 12 10= OC 3 33=OC 768	For WDM OCHCC
PDH	LR_T1_and_DS1_1_5M LR_T3_and_DS3_45M LR_E1_2M LR_E3_34M		For WDM OCHCC Only on ONS15454 SDH
Ethernet	LR_Fast_Ethernet LR_Gigabit_Ethernet LR_Ten_Gigabit_Ethernet LR_Forty_Gigabit_Ethernet	38=FE 11=10Gb/s E 12=1Gb/s E 89=40Gb/s E 1=2.5Gb/s FEC 2=2.5Gb/s No FEC 3=10 Gb/s FEC 4=10 Gb/s No FEC 5=Multirate	For WDM OCHCC
OTU	LR_Optical_Channel LR_OCH_10_Gigabit_ITU LR_OCH_40_Gigabit_ITU	35=OCH OTU2 36=OCHOTU3	For WDM OCHCC
Other	LR_PASS_TROUGH	29=Pass-through 6=Equipped not specific	Other

- CrossConnectList_T ccInclusions

  Not supported.

- ResourceList_T neTpInclusions

  Supported. You must provide a list of managed element names or the PTP names at the endpoints of the topological link. For complete manual routing, you must provide the list of all CTPs of the links (OCH trail links, if any) to which the circuit must be routed.

For protected OCHNC and OCHCC circuits, this parameter defines only the inclusion routing constraints for the working path of the circuit. To define the constraints for the protected path, CTM provides the additional info parameters “includedProtectedME” and “excludedProtectedME.”

- boolean fullRoute

  Not supported.

- ResourceList_T neTpSncExclusions

  Supported. CTM expects a list of managed element names or the PTP names at the endpoints of the TopologicalLink to exclude from the SNC route.

  For protected OCHNC and OCHCC circuits, this parameter defines only the exclusion routing constraints for the working path of the circuit. To define the constraints for the protected path, CTM provides the additional info parameters “includedProtectedME” and “excludedProtectedME.”

- globaldefs::NamingAttributesList_T aEnd

  Supported. CTM expects at least one valid CTP or FTP in this list. At most, two valid CTPs can be in the list.

- globaldefs::NamingAttributesList_T zEnd

  Supported. CTM expects at least one valid CTP or FTP in this list. At most, two valid CTPs can be in the list.

- globaldefs::NVSList_T additionalCreationInfo

  This list can be used to provide additional parameters for SNC creation. The following table provides additional CTM-supported parameters for SNC creation. For more information, refer to the ONS 15454 SONET user documentation.

Table 3-4 CTM-Supported Parameters for SNC Creation

Name	Value	Description
upsrRevertiveMode	Revertive or nonrevertive	If the value is set to revertive, traffic reverts back to the working path when the conditions that diverted it to the protect path are repaired; otherwise, traffic remains on the protect path after the switch.
upsrRevertiveTime	Valid float value from 0.5 to 12, in increments of 12	If the SNC is revertive, you must set this parameter. This is the amount of time that elapses before traffic reverts back to the working path. The default value is 5 minutes. **Note** CTM does not validate this value.
upsrSfBerLevel	3, 4, or 5	Set the thresholds for path protection, path-level signal failure, and bit error rate (only for STS SNCs). **Note** CTM does not validate this value.
upsrSdBerLevel	5, 6, 7, 8, 9	Set the thresholds for path protection, path-level signal degrade, and bit error rate (only for STS SNCs). **Note** CTM does not validate this value.
upsrSwitchOnPdiP	True or false	Set this to true if you want traffic to switch when an STS payload defect indicator is received (only for STS SNCs).
goAndReturnRouting	True or false	Set this to true if you want to provision SNCP/path-protection routes in Go and Return fashion (ITU-T G.841); otherwise, SNCP/path-protection routes are provisioned in unidirectional fashion following GR-1400.

Table 3-4 CTM-Supported Parameters for SNC Creation (continued)

Name	Value	Description
sncDescription	String	Description of the SNC. This value is represented in the userLabel field of the SubnetworkConnection_T structure.
VLAN	VLAN ID	NMS must provide one or more VLAN names. The value indicates the VLAN for CTM to use for SNC creation on Ethernet equipment. An exception is thrown if the VLAN specified does not exist in the multilayer subnetwork of the source TP.
CustomerID	String	User-specified string representing the customer ID.
ServiceID	String	User-specified string representing the service ID.
sncSize	Int	User-specified size for WDM SNCs. Values from 1 to 6 apply to OCHNC and OCH trail SNCs; the default value is 6. Values from 7 to 29 apply to OCHCC SNCs; the value corresponds to that of the layerRate field.
createTunnelOnTransitNodes	True or false	Applicable for VT and VC SNCs with one or more transit nodes that do not have a tunnel or the existing tunnel is exhausted. This parameter specifies whether to create a tunnel on the transit nodes. For SNCs without transit nodes or with an existing tunnel on a transit node that has available bandwidth, this value is ignored. If the value specified is true, a tunnel is created and the SNC is routed using this tunnel. If the value specified is false, a new tunnel is not created. In this case, if a tunnel that is not full is available on the transit node, the circuit is routed through the tunnel; otherwise, the circuit is not routed through a tunnel.
createVTAggregation	True or false	Applicable for VT or VC SNCs with one or more transit nodes that do not have a VAP circuit or if the existing VAP circuit is exhausted. This parameter specifies whether to create a new VAP circuit on the transit nodes. If the value specified is true, a VAP circuit is created and SNC is routed using this new VAP circuit. If the value specified is false, a new VAP circuit is not created.
sourceIsSTSGrooming	True or false	If the value is true, the source drop point of the VT or VC circuit is STS groomed for the VAP circuit. **Note** This attribute is valid only when the createVTAggregation value is set to true. **Note** sourceIsSTSGrooming and destinationIsSTSGrooming cannot both be set to true.
destinationIsSTSGrooming	True or false	If the value is true, the destination drop point of the VT or VC circuit is STS groomed for the VAP circuit. **Note** This attribute is valid only when the createVTAggregation value is set to true. **Note** sourceIsSTSGrooming and destinationIsSTSGrooming cannot both be set to true.

Table 3-4 CTM-Supported Parameters for SNC Creation (continued)

Name	Value	Description
vapDestination	String	You can specify the string (ME name) representing the VAP destination drop point.
usePortlessTransmux	True or false	If the value is true, the portless transmux card is used for the circuit. The NMS provides information for portlessTransmuxMeName, portlessTransmuxSlot, and isSourceDS3MappedSTS; otherwise, an INVALID INPUT exception is thrown. If the value is false, the portless transmux card is not used as the intermediate point for the end-to-end STS-1 circuit.
portlessTransmuxMeName	String	ME name containing the high-density portless transmux card, which is used as a circuit constraint. **Note** This attribute is applicable only if usePortlessTransmux is true.
portlessTransmuxSlot	"/rack=1/shelf=1/slot=[1..6][12..17]"	NMS provides the physical location of the high-density portless transmux card. **Note** Information is required up to the slot level. This attribute is applicable only if usePortlessTransmux is true.
isSourceDS3MappedSTS	True or false	Indicates whether the source or destination is DS3 mapped. **Note** This attribute is applicable only if usePortlessTransmux is true.
vcatMemberId	Integer, 1 to 256	Used for VCAT SNC creation to identify every VCAT member L1 SNC uniquely. For normal CCAT circuits, this value is ignored, if specified.
timeSlotRestriction	Integer, 1 to 192 for SONET and 1 to 64 for SDH.	You can select a specific STS/VC-4 when using autoroute with constraints. The STS/VC-4 is maintained around the ring. If the same STS/VC-4 is not available for the entire circuit route, a warning message is displayed.

includedProtectedME	String	The parameter must match the name of any ManagedElement which you have requested to be included in the protected path of an OCHNC or OCHCC circuit.
		For example:
		`EMS = Cisco Systems/CTM`
		`ManagedElement = myNE`
		Add the following parameter to configure a routing constraint:
		`includedProtectedME = myNE`

Table 3-4 CTM-Supported Parameters for SNC Creation (continued)

Name	Value	Description
excludedProtectedME	String	Must match the name of any ManagedElement which you have requested to be excluded from the protected path of an OCHNC or OCHCC circuit. See the parameter includedProtectedME for the scenario.
overlayRing	String	Values can be: • true—Indicates that an Overlay Ring SNC is allowed. You can create an Overlay Ring SNC only by using CTP Inclusions to pass the complete route from the SNC to CTM Server. • false—Indicates that an Overlay Ring SNC is not allowed.

CTM does not support tolerableImpact (GradesOfImpact_T). Circuits created on the ONS 15327 and ONS 15454 do not impact existing traffic.

CTM does not support emsFreedomLevel (EMSFreedomLevel_T). CTM has full control of SNC creation.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when a field of createData is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when fields of createData reference objects that do not
exist.

EXCPT_UNABLE_TO_COMPLY – Raised if the SNC cannot be created and activated because CTP
conflicts with other SNC or NE does not support this SNC creation. This exception is also
raised when CTM cannot meet protection levels or there is no bandwidth available.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents creation of the SNC.

EXCPT_USERLABEL_IN_USE - Raised when the user label uniqueness constraint is not met.
```

Compliance

CTM-defined.

3.9.17 CreateAndActivateSNC

Synopsis

```
void createAndActivateSNCFromUserLabel(
        in subnetworkConnection::SNCCreateData_T createData,
        in subnetworkConnection::GradesOfImpact_T tolerableImpact,
        in EMSFreedomLevel_T emsFreedomLevel,
        inout subnetworkConnection::TPDataList_T tpsToModify,
        out subnetworkConnection::SubnetworkConnection_T theSNC,
        out string errorReason)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface is identical to 3.9.16 createAndActivateSNCFromUserLabel, page 3-138, except that CTM assigns the name of the created subnetwork connection. The userLabel field in SNCCreateData_T can remain blank. Even if the user label is specified, CTM ignores it and assigns a user label to the SNC.

Compliance

TMF-defined.

3.9.18 (Deprecated) createAndActivateProtectedOchccFromUserLabel

Synopsis

```
void createAndActivateProtectedOchccFromUserLabel(
in subnetworkConnection::SNCCreateData_T createData,
in OCHCCProtectedConstraints_T  ochccProtectedConstraints,
      in subnetworkConnection::GradesOfImpact_T tolerableImpact,
      in EMSFreedomLevel_T emsFreedomLevel,
      inout subnetworkConnection::TPDataList_T tpsToModify,
      out subnetworkConnection::SubnetworkConnection_T theSNC,
      out string errorReason)
    raises(globaldefs::ProcessingFailureException);
```

where:

```
  struct OCHCCProtectedConstraints_T {
    subnetworkConnection::ResourceList_T neTpProtectedInclusions;
    subnetworkConnection::ResourceList_T neTpSncProtectedExclusions;
  };
```

Description

This interface has the same behavior as createAndActivateSNCFromUserLabel, but is used only to create protected OCHCC circuits when it is necessary to specify route constraints for the protected path.

Note This API has been deprecated from CTM R9.0.

Parameters

Name	Type	Input/Output	Description
ochccProtectedConstraints	multiLayerSubnetwork:: OCHCCProtectedConstraints_T	In	Structure containing ME include/exclude route constraints.
other	—	—	Same as for createAndActivateSNCFromUserLabel.

CTM supports the following layer rates:

- ONS 15454 SONET: LR_OCH_Transport_Unit
- ONS 15454 SDH: LR_OCH_Transport_Unit

Compliance

CTM-defined.

3.9.19 addDrops

Synopsis

```
void addDrops(
        in globaldefs::NamingAttributes_T sncName,
        in globaldefs::NamingAttributesList_T zEnd)
    raises(globaldefs::ProcessingFailureException);
```

Description

This proprietary interface supports SNCs with multiple drops for the ONS 15310, ONS 15327, and ONS 15454. There are two steps to create an SNC with multiple drops:

Step 1 Invoke the operation described in 3.9.16 createAndActivateSNCFromUserLabel, page 3-138 to create a unidirectional SNC between two nodes.

Step 2 Invoke this operation to add new drop points to an existing SNC.

Note

- The layer rate must match the SNC layer rate. This interface performs on a best-effort basis. If CTM fails to add one or more TPs to the SNC, it generates an UNABLE_TO_COMPLY exception. The errorReason contains a list of CTPs that failed.
- Depending on the card type, the terminology is *CTP* or *FTP*. For example, termination points on CE-MR-6 cards are CTPs when the cards are configured for manual mode. For CE-MR-6 cards configured for automatic mode, the termination points are FTPs.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the subnetwork connection to which to add termination points (either CTPs or FTPs).
zEnd	globaldefs::NamingAttributesList_T	In	List of additional termination points (either CTPs or FTPs) to add to the SNC.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not refer to an SNC object, or layer rate
of any CTP does not match the SNC layer.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised if the CTP is in use by other SNC. This exception is also
raised when there is no bandwidth available.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost, which
prevents the deactivation of the SNC.
```

Compliance

CTM-defined.

3.9.20 deactivateAndDeleteSNC

Synopsis

```
void deactivateAndDeleteSNC(
in globaldefs::NamingAttributes_T sncName,
      in subnetworkConnection::GradesOfImpact_T tolerableImpact,
      in EMSFreedomLevel_T emsFreedomLevel,
      inout subnetworkConnection::TPDataList_T tpsToModify,
      out subnetworkConnection::SubnetworkConnection_T theSNC,
      out string errorReason)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a way to deactivate and then delete a subnetwork connection in one operation.

CTM removes resources allocated to this SNC from each node. Successful completion of this asynchronous operation guarantees only delivery of the deletion request to CTM. The NMS waits for an OBJECT_DELETION event for the SNC.

CTM performs this operation in two steps:

1. The first step is validation. This is performed synchronously. If successful, this operation returns successfully. If validation fails, the NMS receives an exception.
2. After a successful validation, the second step is the actual deletion of resources from all NEs. This step is performed asynchronously. CTM changes the SNC state to SNCS_DELETING and sends a notification to the NMS. Upon successful completion, CTM generates an OBJECT_DELETION event. If CTM loses connectivity to one or more nodes on the SNC, it aborts the cleanup and generates an SNCS_PARTIAL state change event. The NMS must reinvoke this method when connectivity to the node is restored. If the deletion fails but the state remains SNCS_ACTIVE, CTM generates an AVC event. These SNC state change events are generated as part of the CTM operation only. If another interface is used, CTM cannot generate these events.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the subnetwork connection to deactivate and delete.
tolerableImpact	subnetworkConnection::GradesOfImpact_T	In	CTM ignores this parameter.
emsFreedomLevel	EMSFreedomLevel_T	In	CTM ignores this parameter.
tpsToModify	subnetworkConnection::TPDataList_T	In/out	CTM ignores this parameter.
theSNC	subnetworkConnection::SubnetworkConnection_T	Out	Deactivated and deleted subnetwork connection.
errorReason	string	Out	Specifies the deactivation and/or deletion error(s), if any.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not refer to an SNC object, or any field in
tpsToModify is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the deactivation of the SNC.
```

Compliance

TMF-defined.

3.9.21 getVCATSNC

Synopsis

```
void MultiLayerSubnetworkMgr_I::getVCATSNC(
        in globaldefs::NamingAttributes_T vcatName,
        out subnetworkConnection::VCAT_T vcat)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the VCAT structure for the VCAT SNC whose name is supplied as an argument. The VCAT SNCs are supported on the following equipment types:

- ML equipment (ML1000 and ML100T)
- FCMR
- ADM_10G

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.
vcat	subnetworkConnection::VCAT_T	Out	VCAT structure.

CTM support for VCAT_T is as follows:

- globaldefs::NamingAttributes_T vcatName

  Supported. This field will contain two tuples. The first tuple contains the value for the EMS. The second tuple contains the VCAT name, which is the same as the nativeEMSName. The value of the name field in the last tuple is VCATSubnetworkConnection.

- string userLabel

  Supported.

- string nativeEMSName

  Supported. Name supplied by the user at creation time will be reported.

- string owner

Not supported.

- VCATState_T vcatState

  Supported. The state of the VCAT circuit is derived from the states of the member circuits in the same way as it is derived for CCAT SNCs. The values supported are:

  - VCATState_T.VCATS_NONEXISTENT
  - VCATState_T.VCATS_PENDING
  - VCATState_T.VCATS_ACTIVE
  - VCATState_T.VCATS_PARTIAL
  - VCATState_T.VCATS_DELETING

- globaldefs::ConnectionDirection_T direction

  Supported. This field represents the direction of the VCAT and not the member L1 SNCs.

  CTM supports CD_UNI (unidirectional) and CD_BI (bidirectional). Current release supports only CD_BI.

- boolean isSymmetrical

  Supported. Current release supports only symmetric VCAT SNCs.

  - True if the rate/capacity is same in both directions.
  - False if the rate/capacity is different in each direction.

- transmissionParameters::VCATLayerRate_T txRate

  Supported. This field represents the total capacity the VCAT SNC can carry in the forward direction. If the VCAT is symmetrical, it represents the capacity both in the forward and backward directions. To represent the total capacity of a VCAT SNC, VCATLayerRate_T contains the following information:

  - LayerRate_T memberRate: Layer rate of the member SNCs.
  - Unsigned long numberOfMembers: Number of members for the given VCAT SNC.

  For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- transmissionParameters::VCATLayerRate_T rxRate

  Supported. This field represents the total capacity the VCAT SNC can carry in the backward direction. If the VCAT is symmetrical, this field is ignored. For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- transmissionParameters::LayerRate_T memberRate

  Supported. This field represents the rate of each member circuit.

- globaldefs::NamingAttributes_T aEnd

  Supported. This is an FTP representing the port at the aEnd.

- globaldefs::NamingAttributes_T zEnd

  Supported. This is an FTP representing the port at the zEnd. This field is empty for open-ended VCAT.

- SNCType_T sncType

  Not supported.

- subnetworkConnection::LCASMode_T mode

  Supported. CTM supports the following values:

  - LCASMode_T. LCASMODE _NONE
  - LCASMode_T. LCASMODE _SWLCAS
  - LCASMode_T. LCASMODE _LCAS
- subnetworkConnection::RoutingMode_T routingMode

  Supported. CTM supports the following values:
  - RoutingMode_T. ROUTE_COMMONFIBER
  - RoutingMode_T. ROUTE_INDEPENDENTFIBER
- globaldefs::NVSList_T additionalInfo

  Supported. CustomerID and ServiceID are reported if they are available. The openVcat field reports values of true or false to show whether the SNC is an open-ended VCAT.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not refer to a VCAT SNC object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.
```

Compliance

CTM-defined.

3.9.22 getAllVCATSNCs

Synopsis

```
void MultiLayerSubnetworkMgr_I::getAllVCATSNCs(
        in globaldefs::NamingAttributes_T subnetName,
        in transmissionParameters::LayerRateList_T connectionRateList,
        in unsigned long how_many,
        out subnetworkConnection::VCATList_T vcatList,
        out subnetworkConnection::VCATIterator_I vcatIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns all the VCAT SNCs in the specified multilayer subnetwork. This interface returns only the VCAT SNCs.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
connectionLayerRateList	transmissionParameters::VCATLayerRateList_T	In	List of the VCAT SNCs to report with specified number of members and memberRate. If an empty list is specified, all VCAT SNCs of bandwidth are reported.
how_many	unsigned long	In	Maximum number of SNCs to report in the first batch.

Name	Type	Input/Output	Description
vcatList	subnetworkConnection::VCATList_T	Out	First batch of VCAT SNCs.
vcatIt	subnetworkConnection::VCATIterator_I	Out	Iterator to retrieve the remaining VCAT SNCs.

For details about the VCAT_T structure, see 3.9.21 getVCATSNC, page 3-150.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.9.23 getAllVCATSNCNames

Synopsis

```
void MultiLayerSubnetworkMgr_I::getAllVCATSNCNames(
        in globaldefs::NamingAttributes_T subnetName,
        in transmissionParameters::LayerRateList_T connectionRateList,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the names of all the VCAT SNCs in the specified multilayer subnetwork. This interface returns only the VCAT SNC names.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnetwork.
connectionLayerRateList	transmissionParameters::VCATLayerRateList_T	In	List of the VCAT SNCs to report with specified number of members and member rate. If an empty list is specified, all VCAT SNCs of bandwidth are reported.
how_many	unsigned long	In	Maximum number of SNCs to report in the first batch.

Name	Type	Input/Output	Description
nameList	globaldefs::NamingAttributesList_T	Out	First batch of VCAT SNC names.
nameIt	globaldefs::NamingAttributesIterator_T	Out	Iterator to retrieve the remaining VCAT SNC names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetName does not reference a multiLayerSubnetwork
object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.9.24 getAllVCATMemberSNCs

Synopsis

```
void MultiLayerSubnetworkMgr_I::getAllMemberVCATSNCs(
        in globaldefs::NamingAttributes_T vcatName,
        in boolean isTransmit,
        in unsigned long how_many,
        out subnetworkConnection::SubnetworkConnectionList_T sncList,
        out subnetworkConnection::SubnetworkConnectionIterator_I sncIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns all of the member SNCs for the VCAT SNC.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.
isTransmit	Boolean	In	If the isTransmit boolean is set to true, only transmit-side member SNCs are returned. If set to false, only receive-side member SNCs are returned. If the VCAT is symmetrical, the isTransmit value is ignored and all bidirectional SNCs are returned.
how_many	unsigned long	In	Maximum number of SNCs to report in the first batch.
sncList	subnetworkConnection::SubnetworkConnectionList_T	Out	First batch of SNCs.
sncIt	subnetworkConnection::SNCIterator_I	Out	Iterator to retrieve the remaining SNCs.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not reference a VCAT object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.9.25 getAllVCATMemberSNCNames

Synopsis

```
void MultiLayerSubnetworkMgr_I::getAllVCATMemberSNCNames(
        in globaldefs::NamingAttributes_T vcatName,
        boolean isTransmit,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the names of all member SNCs for a given VCAT SNC.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.
isTransmit	boolean	In	If the isTransmit boolean is set to true, only transmit-side member SNCs are returned. If set to false, only receive-side member SNCs are returned. If the VCAT is symmetrical, the isTransmit value is ignored and all bidirectional SNCs are returned.
how_many	unsigned long	In	Maximum number of member VCAT SNCs to report in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of member VCAT SNC names.
nameIt	globaldefs::NamingAttributesIterator_T	Out	Iterator to retrieve the remaining member VCAT SNC names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not reference a VCAT object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.
```

```
EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.9.26 createVCAT

Synopsis

```
void MultiLayerSubnetworkMgr_I::createVCAT(
        in subnetworkConnection::VCATSNCCreateData_T createData,
        out subnetworkConnection::VCAT_T vcat,
        out string errorReason)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface attempts to create a VCAT SNC.

CTM generates an object creation event for the VCAT SNC that was created and for the individual Layer 1 SNCs.

Parameters

Name	Type	Input/Output	Description
createData	subnetworkConnection::VCATSNCCreateData_T	In	Structure describing the VCAT SNC to create.
vcat	subnetworkConnection::VCAT_T	Out	VCAT SNC created.
errorReason	String	Out	Reason for the failure.

CTM support for VCATSNCCreateData_T structure is as follows:

- string userLabel

  Supported. Value assigned to this field is the name of the new VCAT.

- boolean forceUniqueness

  CTM guarantees the uniqueness of all VCATs created by CTM. CTM does not guarantee uniqueness if any other tool is used for VCAT SNC creation.

- string owner

  Not supported.

- globaldefs::ConnectionDirection_T direction

  CTM supports only CD_BI (bidirectional).

- boolean isSymmetrical

  Supported. In the current release, only symmetric VCAT SNCs are supported. If set to true, all members are bidirectional L1 SNCs, and the VCAT layer rate is the same for both directions. If set to false, all members are unidirectional L1 SNCs, and the supported VCAT layer rate on either direction can be different.

- subnetworkConnection::LCASMode_T mode

  Supported. CTM supports the following values:

  - LCASMode_T.MODE_NONE
  - LCASMode_T.MODE_SWLCAS
  - LCASMode_T.MODE_LCAS
- subnetworkConnection::RoutingMode_T routingMode

  Supported. CTM supports the following values:
  - RoutingMode_T.COMMONFIBER_ROUTED
  - RoutingMode_T.INDEPENDENTFIBER_ROUTED
- subnetworkConnection::TPData_T aEnd

  Supported. CTM expects this field to be an FTP representing the information up to port level of the card involved in VCAT SNC.
- subnetworkConnection::TPData_T zEnd

  Supported. CTM expects this field to be an FTP representing the information up to the port level of the card involved in the VCAT SNC. This field is empty for open-ended VCAT.
- subnetworkConnection::SNCCreateDataList_T sncCreateDataList

  Supported. At least one member L1 SNC must be created at the time of VCAT SNC creation. CTM expects user to specify the create data information required to create the L1 SNC(s) in this data structure. For complete information on SNCCreateData_T structure, see 3.9.16 createAndActivateSNCFromUserLabel, page 3-138. For every L1 SNC, you must specify an additionalCreationParamemter vcatMemberId, which must contain a unique value from 1 to 256.
- globaldefs::NVSList_T additionalCreationInfo

  This list can be used to provide additional parameters for VCAT creation. CTM supports the following values:

Name	Value	Description
vcatDescription	String	Description of the VCAT.
CustomerID	String	User-specified string representing the customer ID.
ServiceID	String	User-specified string representing the service ID.
useExtendedBuffer	Boolean (True/False)	For differential delay.
openVcat	Boolean (True/False)	Indicates that an open-ended VCAT must be created.

- subnetworkConnection::ResourceList_T neTpInclusions

  Supported. You must provide a list of managed element names or the PTP names at the endpoints of the topological link. For complete manual routing, you must provide the list of all CTPs of the links (OCH trail links, if any) to which the circuit must be routed.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when a field of createData is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when fields of createData reference objects that do not
exist.
```

```
EXCPT_UNABLE_TO_COMPLY - Raised if the VCAT SNC cannot be created and activated because
FTP conflicts with another VCAT SNC or NE does not support this VCAT SNC creation. This
exception is also raised when CTM cannot meet protection levels or there is no bandwidth
available.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents creation of the VCAT SNC.

EXCPT_USERLABEL_IN_USE - Raised when the user label uniqueness constraint is not met.
```

Compliance

CTM-defined.

3.9.27 deleteVCAT

Synopsis

```
void deleteVCAT(
        in globaldefs::NamingAttributes_T vcatName)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface deletes the VCAT specified in the input. CTM generates an object deletion event for the VCAT SNC that was deleted and for the individual Layer 1 SNCs that were deleted.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not reference a VCAT object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when deletion of the VCAT SNC fails.
```

Compliance

CTM-defined.

3.9.28 addMemberSNCsToVCAT

Synopsis

```
void addMemberSNCsToVCAT(
        in globaldefs::NamingAttributes_T vcatName,
        in subnetworkConnection::SNCCreateDataList_T sncData,
        out subnetworkConnection::VCAT_T vcat)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface adds a member SNC to the VCAT that is specified in the input. CTM generates an attribute value change event for the VCAT SNC that was modified due to the addition of the new SNC(s), and an object creation event for the member L1 SNCs that were created.

For each new member, you must specify an additional creation parameter called "vcatMemberId," which is a unique number from 1 to 256. The value that you choose cannot be in use by any other members that belong to the VCAT SNC.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.
sncData	subnetworkConnection::SNCCreateDataList_T	In	Structure describing the SNC(s) to create and add to the VCAT.
vcat	subnetworkConnection::VCAT_T	Out	The augmented VCAT SNC.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not reference a VCAT object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when addition of a member SNC to the VCAT fails.
```

Compliance

CTM-defined.

3.9.29 deleteMemberSNCsFromVCAT

Synopsis

```
void deleteMemberSNCsFromVCAT(
        in globaldefs::NamingAttributes_T vcatName,
        in globaldefs::NamingAttributesList_T sncName,
        out subnetworkConnection::VCAT_T vcat)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface deletes a member SNC from the VCAT specified in the input. CTM generates an attribute value change event for the VCAT SNC that was modified due to the deletion of the member SNC, and an object deletion event for the member L1 SNC that was deleted.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributesList_T	In	Name of the SNC(s) to delete and remove from the VCAT.
vcat	subnetworkConnection::VCAT_T	Out	Modified VCAT SNC.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName or sncName does not reference a VCAT/SNC
object.

EXCPT_ENTITY_NOT_FOUND - Raised when vcatName or sncName references an object that does
not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when deletion of member SNC(s) from the VCAT fails.
```

Compliance

CTM-defined.

3.9.30 deleteAllMemberSNCFromVCAT

Synopsis

```
void deleteAllMemberSNCFromVCAT(
        in globaldefs::NamingAttributes_T vcatName,
        in boolean isForward)
     raises( globaldefs::ProcessingFailureException );
```

Description

This interface deletes all member SNCs in the VCAT group that is specified during input. If boolean value is set to true, all member SNCs in the transmit direction are deleted. If boolean value is set to false, all member SNCs in the receive direction are deleted. For symmetrical VCAT, this value is ignored; all member circuits in both transmit and receive directions are deleted. CTM generates an attribute value change for the VCAT SNC modified due to the deletion of member SNC(s), and an object deletion event for the member L1 SNCs deleted.

Parameters

Name	Type	Input/Output	Description
vcatName	globaldefs::NamingAttributes_T	In	Name of the VCAT SNC.
isForward	Boolean	In	Boolean variable used to determine the direction in which the member SNCs of the VCAT are deleted.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when vcatName does not reference a VCAT object.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when vcatName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when deletion of member SNC(s) from the VCAT fails.
```

Compliance

CTM-defined.

3.9.31 getValidSubnetworkConnectionsForMerge

Synopsis

```
void getValidSubnetworkConnectionsForUpgrade(
        in multiLayerSubnetwork::SncUpgradeInfo_T details,
        in multiLayerSubnetwork::SncFilterCriteria_T filterCriteria,
        in globaldefs::NamingAttributes_T masterSncName,
        in globaldefs::NamingAttributesList_T candidateSncList,
        in unsigned long how_many,
        out subnetworkConnection::SubnetworkConnectionList_T mergeableSncList,
        out subnetworkConnection::SNCIterator_I sncIt )
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface is used to get the list of SNCs for the Merge operation. The SncFilterCriteria specifies the SNCs that are requested. These SNCs are lined up. The master SNC name specifies the name of the SNC with which the candidates will merge. The how_many parameter provides the list of SNCs that will be returned.

Note The valid SncUpgradeInfo value supported by CTM GateWay/CORBA is Merge. The valid SncFilterCriteria value supported by CTM GateWay/CORBA is LINED_UP. The available values that will be supported in the future are listed in the ENUM definitions in the following table.

Parameters

Name	Type	Input/Output	Description
details	multiLayerSubnetwork::SncUpgradeInfo_T	In	Upgrade info for which the list of SNCs is requested. Available values are: • Merge • Reconfigure **Note** CTM GateWay/CORBA R9.2 supports only Merge.
filterCriteria	multiLayerSubnetwork::SncFilterCriteria_T	In	The filter criteria can be used to specify if the requested SNCs are LINED_UP or nonfiltered. **Note** CTM GateWay/CORBA R9.2 supports only LINED_UP.
masterSncName	globaldefs::NamingAttributes_T	In	This parameter is used to specify the name of the SNC to merge or upgrade with the following candidate SNCs.
candidateSncList	globaldefs::NamingAttributesList_T	In	This parameter gives a list of SNCs to merge with the master SNC.

Name	Type	Input/Output	Description
how_many	unsigned long	In	This parameter specifies the number of SNCs to pass and the SNCs that are valid for the merge operation with the master SNC.
mergeableSncList	subnetworkConnection::SubnetworkConnectionList_T	Out	This parameter gives the list of SNCs that qualify for a merge.
sncIt	subnetworkConnection::SNCIterator_I	Out	This parameter is used to iterate the list of SNCs returned by the API.

```
enum SncUpgradeType_T {
        MERGE,
        RECONFIGURE
    };

enum SncOperationType_T {
        STRICT_MERGE,
        LOOSE_MERGE,
        NONE
    };
enum SncFilterCriteria_T {
        LINED_UP,
        NO_FILTER
    };

struct SncUpgradeInfo_T {
        SncUpgradeType_T upgradeType;
        SncOperationType_T operationType;
    };
```

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.9.32 upgrade

Synopsis

```
void upgrade(
       in multiLayerSubnetwork::SncUpgradeInfo_T details,
       in globaldefs::NamingAttributes_T masterSncName,
       in globaldefs::NamingAttributesList_T upgradableSncNameList,
       out string resultString )
       raises(globaldefs::ProcessingFailureException);
```

Description

This interface is used to merge or configure a list of SNCs to the SNC reported in the masterSncName. This interface supports STRICT MERGE and LOOSE MERGE as operation types from the OSS. The LOOSE MERGE option in SncUpgradeInfo_T is available only if a strict operation failed and the OSS receives a processing failure exception with EXCPT_STRICT_MERGE_FAILED.

There is no restriction in CTM GateWay/CORBA to prevent a LOOSE MERGE operation from the OSS during a normal call.

Parameters

Name	Type	Input/Output	Description
details	multiLayerSubnetwork::SncUpgradeInfo_T	in	Upgrade info for which the list of SNCs are requested. Available values are: • Merge • Reconfigure
masterSncName	globaldefs::NamingAttributes_T	In	This parameter is used to specify the name of the SNC to merge or reconfigure with the following candidate SNCs.
upgradableSncNameList	globaldefs::NamingAttributesList_T	In	This parameter gives the list of SNCs to merge or reconfigure with the master SNC.
resultString	string	Out	This parameter gives the result of the operation in human-readable form.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_STRICT_MERGE_FAILED – Raised when a strict merge operation from the OSS fails. The
OSS uses this exception to issue a loose merge call.
```

Compliance

CTM-defined.

3.9.33 createRoll

Synopsis

```
void createRoll(
        in multiLayerSubnetwork::RollCreateData_T rollCreateData)
     raises(globaldefs::ProcessingFailureException);
```

Description

CTM supports roll creation on optical CTPs. Electrical CTPs and FTPs are not supported for bridge and roll creation.

CTM generates an Object Creation event for the roll created and subnetwork connections created as part of the roll creation. The subnetwork connections that are created as part of the roll creation will be named as nativeEMSName of the SNC appended with _ROLL** where ** is a number.

The creation of bridge and roll is an asynchronous process. If the operation fails, CTM cleans up all resources. For example, if CTM fails to create a bridge and roll on one node, it also cleans up resources on other nodes.

Parameters

Name	Type	Input/Output	Description
rollCreateData	multiLayerSubnetwork::RollCreateData_T	In	Structure describing the roll to create.

For details about the Roll_T structure, see 3.8.28 getAllRolls, page 3-111.

CTM support for RollCreateData_T is as follows:

- multiLayerSubnetwork::RollType_T rollType

  Supported.

- multiLayerSubnetwork::RollMode_T rollMode

  Supported.

- globaldefs::NamingAttributesList_T sncNames

  Supported. This refers to the name of the SNC to roll partially or fully. If two SNCs are specified, the first SNC will be the RollFromSNC and the second SNC will be the RollToSNC.

- globaldefs::NamingAttributesList_T fixCTPs

  Supported. This refers to the fully qualified name of the CTP of the SNC. This CTP is the fixed point of the roll. In cases where the rollType is DUAL, there will be two fixed points in the roll. The first fixCTP indicates the first fixed point in the roll object and the second fixCTP indicates the second fixed point of the roll.

- globaldefs::NamingAttributesList_T rollFromCTP

  Supported. This refers to the fully qualified name of the CTP of the SNC. This CTP is either the source or destination or any other associated CTP of the SNC. In cases where the rollType is DUAL, the first element in the array indicates the first rollFromCTP point and the second element in the array indicates the second rollFromCTP point.

  Note If the rollFromCTP can be determined from the fixed CTP, the value is ignored.

- globaldefs::NamingAttributesList_T rollToCTP

  Supported. This refers to the fully qualified name of the CTP where the roll occurs. In cases where the rollType is DUAL, the first element in the array indicates the first rollToCTP point and the second element in the array indicates the second rollToCTP point.

- subnetworkConnection::ResourceList_T neTpInclusions

  Supported. CTM expects a list of managed element names or PTP names of the endpoint of a topological link or CTP name pair representing the endpoints of the span to include in the subnetwork connection route. For complete manual routing, CTM expects the list of all CTPs of the links (OCH Trail circuit links, if any) to which the circuit must be routed. If the OSS specifies a CTP pair in the neTpInclusions, the entire path for the subnetwork connection must be specified.

- subnetworkConnection::ResourceList_T neTpSncExclusions

  Supported. CTM expects a list of managed element names or PTP names of the endpoint of a topological link representing the endpoints of the span to exclude in the subnetwork connection route.

- subnetworkConnection::StaticProtectionLevel_T staticProtectionLevel

  CTM supports the following for the subnetwork connection that is created as part of the roll creation:

- UNPROTECTED
- FULLY_PROTECTED
- HIGHLY_PROTECTED

Note CTM does not support other values.

- subnetworkConnection::ProtectionEffort_T protectionEffort

  If the NMS requests unprotected SNC on a BLSR ring, CTM creates the SNC. The same is true for unprotected SNC on 1+1 protection.

  If the NMS requests EFFORT_SAME_OR_WORSE or EFFORT_WHATEVER, CTM expects a value of staticProtectionLevel of UNPROTECTED. These values indicate that CTM must create the PCA.

CTM support for Roll_T is described in 3.8.28 getAllRolls, page 3-111.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when sncName does not refer to an SNC object, or any field in
rollCreateData is invalid, or the data does not conform to roll creation rules.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the deactivation of the SNC.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

Scenarios: Roll Creation

Figure 3-1 Example 1: Roll Source or Destination of an SNC—Part 1

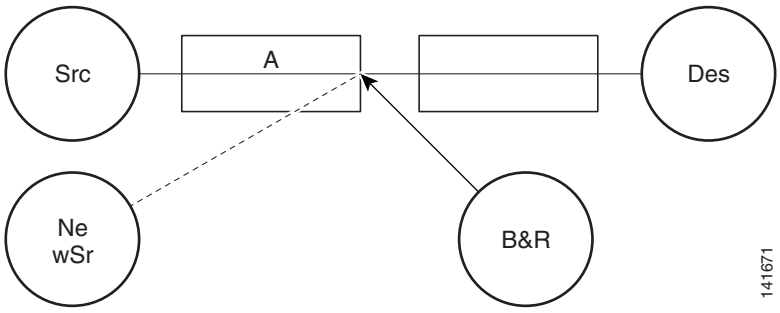


Figure 3-2 Example 2: Roll Source or Destination of an SNC—Part 2

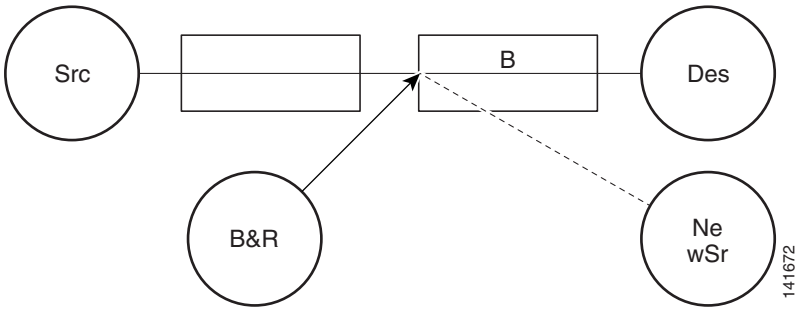
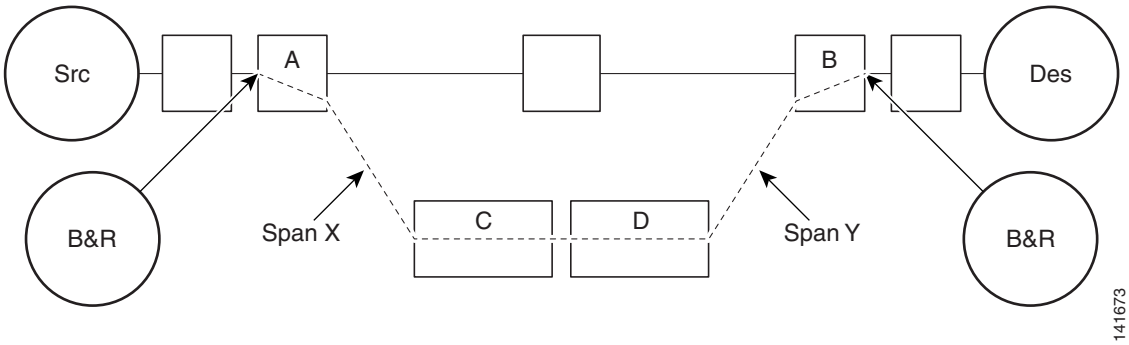


Figure 3-3 Example 3: Roll Intermediate Managed Elements or Topological Links



3.9.34 deleteRoll

Synopsis

```
void deleteRoll(
        in globaldefs::NamingAttributes_T rollname)
    raises(globaldefs::ProcessingFailureException);
```

Description

CTM generates an object deletion event for the deleted roll object. CTM also generates object deletion events for the SNCs that are deleted.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.
```

```
EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the deletion of the roll.
```

Compliance

CTM-defined.

3.9.35 cancelRoll

Synopsis

```
void cancelRoll(
        in globaldefs::NamingAttributes_T rollName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface cancels a roll task when the rollValidSignal is not received. For more information on canceling a roll task, see the network element documentation.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll to cancel.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the cancellation of the roll operation.

EXCPT_UNABLE_TO_COMPLY – Raised when the roll cannot be created.
```

Compliance

CTM-defined.

3.9.36 forceValidSignal

Synopsis

```
void forceValidSignal(
        in globaldefs::NamingAttributes_T rollName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface runs on both automatic and manual rolls.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll on which to force a valid signal.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.9.37 completeRoll

Synopsis

```
void completeRoll(
        in globaldefs::NamingAttributes_T rollName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface must run on manual rolls (RollMode_T.MANUAL).

CTM generates a state change event upon completion of the roll.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll to complete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the deactivation of the SNC.
```

Compliance

CTM-defined.

3.9.38 finishRoll

Synopsis

```
void completeRoll(
        in globaldefs::NamingAttributes_T rollName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface is invoked to finish rolls. When the roll is finished, it is cleared from the rolls list and the new rolled SNC becomes active.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of the roll to finish.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost and this
prevents the deactivation of the SNC.
```

Compliance

CTM-defined.

3.9.39 getAllRolls

Synopsis

```
void getAllRolls(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out multiLayerSubnetwork::RollList_T rollList,
        out multiLayerSubnetwork::RollIterator_I rollIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of rolls within the subnetwork whose name is passed as a parameter.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnet for which all rolls are reported.
how_many	unsigned	In	Maximum number of rolls to report in the first batch.

Name	Type	Input/Output	Description
rollList	multiLayerSubnetwork::RollList_T	Out	First batch of rolls.
rollIt	multiLayerSubnetwork::RollIterator_I	Out	Iterator to retrieve remaining rolls.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetwork name does not refer to a subnetwork object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetwork references an object that does not exist.
```

Compliance

CTM-defined.

3.9.40 getAllRollNames

Synopsis

```
void getAllRollNames(
        in globaldefs::NamingAttributes_T subnetName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of roll names within the subnetwork whose name is passed as a parameter.

Parameters

Name	Type	Input/Output	Description
subnetName	globaldefs::NamingAttributes_T	In	Name of the subnet for which all rolls are reported.
how_many	unsigned	In	Maximum number of rolls to report in the first batch.
nameList	globaldefs::NamingAttributesList	Out	First batch of roll names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining roll names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when subnetwork name does not refer to a subnetwork object.

EXCPT_ENTITY_NOT_FOUND - Raised when subnetwork references an object that does not exist.
```

Compliance

CTM-defined.

3.9.41 getAssociatedRollNamesForSNC

Synopsis

```
void getAssociatedRollNamesForSNC(
        in globaldefs::NamingAttributes_T sncName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList rollNameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns all the rolls associated with the SNC.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the SNC whose associated roll objects are required.
how_many	unsigned long	In	Maximum number of rolls to report in the first batch.
rollNameList	globaldefs::NamingAttributesList	Out	First batch of roll names.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator to retrieve the remaining roll names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not refer to an SNC object.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an object that does not exist.
```

Compliance

CTM-defined.

3.9.42 getAssociatedSNCNamesForRoll

Synopsis

```
void getAssociatedSNCNamesForRoll(
        in globaldefs::NamingAttributes_T rollName,
        out globaldefs::NamingAttributes_T rollFromSnc,
        out globaldefs::NamingAttributes_T rollToSnc)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the rollFrom SNC name and rollTo SNC name for a roll whose name is passed as a parameter. For circuit bridge-and-roll operations, the CTM client Rolls table contains roll-from circuit and roll-to circuit attributes. The getAssociatedSNCNamesForRoll method returns these attributes for a given circuit roll.

Parameters

Name	Type	Input/Output	Description
rollName	globaldefs::NamingAttributes_T	In	Name of roll whose associated SNC will be reported.
rollFromSnc	globaldefs::NamingAttributes_T	Out	Name of the rollFrom SNC.
rollToSnc	globaldefs::NamingAttributes_T	Out	Name of the rollTo SNC.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when rollName does not refer to a roll object.

EXCPT_ENTITY_NOT_FOUND - Raised when rollName references an object that does not exist.
```

Compliance

CTM-defined.

3.9.43 createAndActivateSNCOnDRI

Synopsis

```
void createAndActivateSNCOnDRI(
        in subnetworkConnection::DRISNCCreateData_T createData,
        in subnetworkConnection::GradesOfImpact_T tolerableImpact,
        in EMSFreedomLevel_T emsFreedomLevel,
        inout subnetworkConnection::TPDataList_T tpsToModify,
        out subnetworkConnection::SubnetworkConnection_T theSNC,
        out string errorReason)
    raises(globaldefs::ProcessingFailureException);
```

Description

This structure was introduced to provide BLSR-DRI protection support. This structure is not defined by TMF. This structure is similar to createAndActivateSNCOnUserLabel (see 3.9.14 getSNCsByUserLabel, page 3-136 for more details) except that a new field, driNode, has been added.

Parameters

Name	Type	Input/Output	Description
createData	subnetworkConnection::DRISNCCreateData_T	In	New structure describing the SNC to create and activate. This is similar to SNCCreateData_T except that DRI-specific information has been added.
tolerableImpact	subnetworkConnection::GradesOfImpact_T	In	Maximum tolerable impact allowed.
emsFreedomLevel	EMSFreedomLevel_T	In	Maximum level of freedom allowed to the EMS to perform the creation and activation.

Name	Type	Input/Output	Description
tpsToModify	subnetworkConnection::TPDataList_T	In/out	CTM ignores this parameter.
theSNC	subnetworkConnection::SubnetworkConnection_T	Out	Resulting SNC, with sncState and name set. The name is set to the user label supplied in createData structure.
errorReason	String	Out	Specifies the creation and/or activation error(s), if any.

The structure of DRISNCCreateData_T is as follows:

```
struct DRISNCCreateData_T
{
    subnetworkConnection::SNCCreateData_T createData,
    subnetworkConnection::DRINodesList_T nodeList,
};

typedef sequence <DRINodes_T> DRINodesList_T;
```

The structure of DRINodes_T is as follows:

```
struct DRINodes_T
{
    globalDefs::DRIType_T type,
    boolean isSecPCA;
    DriNodePairList_T nodePairList;
    globaldefs::NamingAttributesList requiredRings;
}

typedef sequence <DRINodePair_T> DRINodePairList_T;

struct DRINodePair_T
{
    globaldefs::NamingAttributes_T PrimaryME;
    globaldefs::NamingAttributes_T SecondaryME
}
```

CTM support for DriNode_T structure is as follows:

- type

  This field defines integer values that categorize the BLSR-DRI as traditional or nontraditional BLSR-DRI. In CTM type = 0 means it is a dummy node, 1 = traditional. and 2 = nontraditional.

- isSecPCA

  This field indicates whether or not the RIP option is supported.

- DriNodePair_T

  This structure holds the primary and secondary nodes. The sDriNodePair structure is as follows:

  - globaldefs::NamingAttributes_T PrimaryME—Defines the primary DRI node.
  - globaldefs::NamingAttributes_T SecondaryME —Defines the secondary DRI node.
  - globaldefs::NamingAttributesList requiredRings—Holds the ring information. The first name value tuple represents EMS instance and the second one holds the BLSR ring ID.

Compliance

CTM-defined.

3.9.44 addManagedElement

Synopsis

```
void addManagedElement(in MEAdditionData_T meData)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows the NMS to add a new managed element to the specified subnetwork.

- If the operation fails, an exception is returned, and the managed element is not added to the subnetwork.
- If the operation succeeds, a managed element is added to the specified subnetwork. The EMS populates the ME communication state, its supported layer rates, and emsInSyncState when connectivity to the ME is established.

CTM generates an NT_OBJECT_CREATION event upon successful addition of a managed element.

Parameters

Name	Type	Input/Output	Description
meData	MEAdditionData_T	In	Input data required for the addition of the NE.

The structure of MEAdditionalData_T is as follows:

```
struct MEAdditionData_T
{
    globaldefs::NamingAttributes_T managedElementName;
    globaldefs::NamingAttributes_T containingSubnetworkName;
    in string userLabel;
    in string nativeEMSName;
    in string owner;
    in string location;
    in string version;
    in string productName;
    globaldefs::NVSList_T additionalInfo;
};
```

The description of each field is as follows:

- globaldefs::NamingAttributes_T managedElementName—Name of the managed element to add.
  - name="ManagedElement"
  - value=Managed Element
- containingSubnetworkName—Name of the subnetwork that contains the managed element.
  - name="subnetworkName"
  - value=subnetwork name
- string userLabel—User-specified label of the managed element.
- string nativeEMSName—nativeEMSName of the managed element to add.
- string owner—Owner of the managed element.
- string location—Location of the managed element.
- string version—The NMS can specify the ME version; however, this value is overwritten when the ME is put in service and the EMS retrieves the actual version information.

- string productName—Product name of the managed element.
- globaldefs::NVSList_T additionalInfo—Parameters that must be specified to the EMS to add the ME; for example, IP address, ME username, password, and any other vendor-specific proprietary data.

CTM support for addManagedElement is as follows:

- globaldefs::NamingAttributes_T managedElementName: Not supported
- globaldefs::NamingAttributes_T containingSubnetworkName: Supported
- string userLabel: Not supported
- string nativeEMSName: Not supported
- string owner: Not supported
- string location (location of the managed element): Supported
- string version: Not supported
- string productName: Not supported
- globaldefs::NVSList_T additionalInfo: Supported

The structure of Additional Info is as follows:

Name	Value	Description	Mandatory or Optional
NEModel	modelname	NE model name	Mandatory (see the following values)
IPAddress	ipaddress	IP address of the network element	Mandatory (for non-ONS 15216 NEs)
NPName	network partition name	Name of the network partition under which to add the NE	Mandatory
NESYSID	nesysid	NE ID for the NE	Mandatory (for ONS 15216 NEs)
NEUsername	username	Username	Mandatory (for non-ONS 15216 NEs)
NEPassword	password	Password	Mandatory (for non-ONS 15216 NEs)
Description	description	Description	Optional
Community String	private/public	SNMP community string	Mandatory (see the following values)
Vendor	vendor	NE version name	Optional

The parameter name “Community String” in additionalInfo can have the following values:

- For NE types ONS 15216 EDFA2 and ONS 15216 EDFA3, the Community String value is *private*.
- For OADM NE types, the Community String parameter is not required.
- For the remaining NE types, the Community String value is *public*.

The parameter name “NEModel” in additionalInfo can have the following values:

- "CISCO_ONS_15327_UPSR_ADM"
- "CISCO_ONS_15327_TSA_TERMINAL"
- "CISCO_ONS_15327_BLSR_ADM"
- "CISCO_ONS_15327_HYBRID"
- "CISCO_ONS_15327_Other"
- "CISCO_ONS_15327_Unknown"
- "CISCO_ONS_15454_UPSR_ADM"

- "CISCO_ONS_15454_BLSR_ADM"
- "CISCO_ONS_15454_TSA_TERMINAL"
- "CISCO_ONS_15454_OPTICAL_ADM"
- "CISCO_ONS_15454_HUB"
- "CISCO_ONS_15454_DWDM_HYBRID"
- "CISCO_ONS_15454_TERMINAL_HUB_EAST"
- "CISCO_ONS_15454_TERMINAL_HUB_WEST"
- "CISCO_ONS_15454_DWDM_TERMINAL"
- "CISCO_ONS_15454_PASSIVE_LINE_SITE"
- "CISCO_ONS_15454_ACTIVE_LINE_SITE"
- "CISCO_ONS_15454_ROADM"
- "CISCO_ONS_15454_TERMINAL_OADM_EAST"
- "CISCO_ONS_15454_TERMINAL_OADM_WEST"
- "CISCO_ONS_15454_DWDM_HUB_NODE"
- "CISCO_ONS_15454_OSC_REGENERATION_SITE"
- "CISCO_ONS_15454_L_LINE_SITE"
- "CISCO_ONS_15454_L_ROADM"
- "CISCO_ONS_15454_L_TERMINAL"
- "CISCO_ONS_15454_L_HUB"
- "CISCO_ONS_15454_Other"
- "CISCO_ONS_15454_Unknown"
- "CISCO_ONS_15540_ESP"
- "CISCO_ONS_15540_ESPx"
- "CISCO_ONS_15530"
- "CISCO_ONS_15454_SDH_MSSPRING"
- "CISCO_ONS_15454_SDH_SNC_ADM"
- "CISCO_ONS_15454_SDH_TSA_TERMINAL"
- "CISCO_ONS_15454_SDH_TSA_TERMINAL_EAST"
- "CISCO_ONS_15454_SDH_TSA_TERMINAL_WEST"
- "CISCO_ONS_15454_SDH_OPTICAL_ADM"
- "CISCO_ONS_15454_SDH_HUB"
- "CISCO_ONS_15454_SDH_HYBRID"
- "CISCO_ONS_15454_SDH_DWDM_HUB_NODE"
- "CISCO_ONS_15454_SDH_PASSIVE_LINE_SITE"
- "CISCO_ONS_15454_SDH_ACTIVE_LINE_SITE"
- "CISCO_ONS_15454_SDH_LINE_AMPLIFIER"
- "CISCO_ONS_15454_SDH_DWDM_TERMINAL"
- "CISCO_ONS_15454_SDH_ROADM"

- "CISCO_ONS_15454_SDH_L_LINE_SITE"
- "CISCO_ONS_15454_SDH_L_ROADM"
- "CISCO_ONS_15454_SDH_L_TERMINAL"
- "CISCO_ONS_15454_SDH_L_HUB"
- "CISCO_ONS_15454_SDH_Other"
- "CISCO_ONS_15454_SDH_Unknown"
- "CISCO_ONS_15600_UPSR_ADM"
- "CISCO_ONS_15600_TSA_TERMINAL"
- "CISCO_ONS_15600_BLSR_ADM"
- "CISCO_ONS_15600_LINEAR_ADM"
- "CISCO_ONS_15600_DCS"
- "CISCO_ONS_15600_HYBRID"
- "CISCO_ONS_15600_Other"
- "CISCO_ONS_15600_Unknown"
- "CISCO_ONS_15600_SDH_MSSPRING"
- "CISCO_ONS_15600_SDH_TSA_TERMINAL"
- "CISCO_ONS_15600_SDH_LINEAR_ADM"
- "CISCO_ONS_15600_SDH_DCS"
- "CISCO_ONS_15600_SDH_HYBRID"
- "CISCO_ONS_15600_SDH_Other"
- "CISCO_ONS_15600_SDH_Unknown"
- "CISCO_ONS_15305"
- "CISCO_ONS_15305_CTC_SNCP"
- "CISCO_ONS_15305_CTC_Unknown"
- "CISCO_ONS_15305_CTC_Other"
- "Not_Managed_Other_Vendor"
- "CISCO_ONS_15216_EDFA1"
- "CISCO_ONS_15216_EDFA2"
- "CISCO_ONS_15216_EDFA3_POST_AMPLIFIER"
- "CISCO_ONS_15216_EDFA3_LINE_AMPLIFIER"
- "CISCO_ONS_15216_EDFA3_PRE_AMPLIFIER"
- "CISCO_ONS_15216_EDFA3_Other"
- "CISCO_ONS_15216_EDFA3_Unknown"
- "CISCO_ONS_15216_OSC_1510"
- "CISCO_ONS_15216_DCU"
- "CISCO_ONS_15216_FlexLayer"
- "CISCO_ONS_15216_200_GHZ_OADM1"
- "CISCO_ONS_15216_200_GHZ_OADM2"

- "CISCO_ONS_15216_100_GHZ_OADM4"
- "CISCO_ONS_15216_100_GHZ_OADM2"
- "CISCO_ONS_15216_100_GHZ_OADM1"
- "CISCO_ONS_15216_DWDM_Filter_RED_200_GHZ_DWDM_FILTER"
- "CISCO_ONS_15216_DWDM_Filter_BLUE_200_GHZ_DWDM_FILTER"
- "CISCO_ONS_15216_DWDM_Filter_RED_100_GHZ_DWDM_FILTER"
- "CISCO_ONS_15216_DWDM_Filter_BLUE_100_GHZ_DWDM_FILTER"
- "CISCO_ONS_15310_CL_UPSR_ADM"
- "CISCO_ONS_15310_CL_TSA_TERMINAL"
- "CISCO_ONS_15310_CL_BLSR_ADM"
- "CISCO_ONS_15310_CL_HYBRID"
- "CISCO_ONS_15310_CL_Other"
- "CISCO_ONS_15310_CL_Unknown"
- "CISCO_MDS_9100"
- "CISCO_MDS_9200"
- "CISCO_MDS_9500"
- "CISCO_MDS_Other"
- "CISCO_MDS_Unknown"
- "CISCO_MGX_8880_8850"
- "CISCO_ONS_15310_MA_UPSR_ADM"
- "CISCO_ONS_15310_MA_TSA_TERMINAL"
- "CISCO_ONS_15310_MA_BLSR_ADM"
- "CISCO_ONS_15310_MA_HYBRID"
- "CISCO_ONS_15310_MA_Other"
- "CISCO_ONS_15310_MA_Unknown"
- "CISCO_ONS_15310_MA_SDH_UPSR_ADM"
- "CISCO_ONS_15310_MA_SDH_TSA_TERMINAL"
- "CISCO_ONS_15310_MA_SDH_BLSR_ADM"
- "CISCO_ONS_15310_MA_SDH_HYBRID"
- "CISCO_ONS_15310_MA_SDH_Other"
- "CISCO_ONS_15310_MA_SDH_Unknown"

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when containingSubnetworkName does not reference a valid
subnetwork name.

EXCPT_OBJECT_IN_USE - Raised if the managed element already exists.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.
```

Compliance

CTM-defined.

3.9.45 deleteManagedElement

Synopsis

```
void deleteManagedElement (
          in globaldefs::NamingAttributes_T managedElementName)
          raises(globaldefs::ProcessingFailureException)
```

Description

This interface allows the NMS to delete a managed element. CTM generates an NT_OBJECT_DELETION event upon successful deletion of the managed element.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element is not found.

EXCPT_UNABLE_TO_COMPLY - Raised if the EMS is unable to comply.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.
```

Compliance

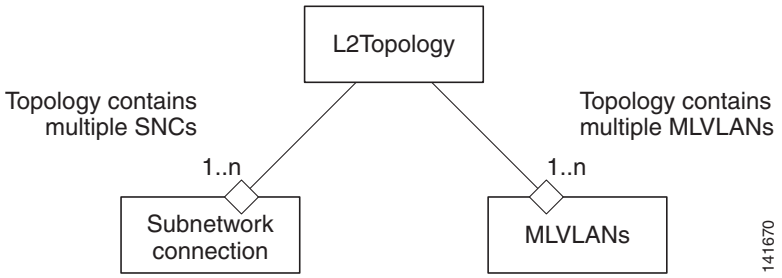
CTM-defined.

3.10 L2 Topology and ML VLAN for ML-Series Ethernet Cards

CTM supports L2 topology and ML VLAN operations for the ML-series Ethernet cards in ONS 15454 SONET and ONS 15454 SDH NEs. You can create an ML VLAN in an L2 topology. An L2 topology has a collection of ML VLANs and underlying subnetwork connections to support the traffic at the optical layer.

The following figure shows the relationship diagram for various L2 service objects.

Figure 3-4 L2 Service Object Relationship Diagram



CTM supports L2 service interfaces for the ONS 15454 SONET and ONS 15454 SDH R4.1 and later.

3.10.1 createMLVLAN

Synopsis

```
void createMLVLAN(in String protocol,in globaldefs::NamingAttributes_T l2TopologyName,
        in multiLayerSubnetwork::MLVLANCreateData_T mlVlanCreateData,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface creates a new ML VLAN for a given L2 topology. An object creation event is generated when an ML VLAN is created successfully. CTM supports a maximum of 4095 ML VLANs for an L2 topology.

Parameters

Name	Type	Input/ Output	Description
protocol	string	In	This VLAN runs the set protocol. If you want to run the default protocol, pass an empty string. Otherwise, you can set different protocols; for example, RSTP. You must set the RSTP protocol on the VLAN to enable RSTP on the port that belongs to the VLAN.
l2TopologyName	globaldefs::NamingAttributes_T	In	The L2 topology name in which this ML VLAN is created.
mlVlanCreateData	multiLayerSubnetwork::MLVLANCreateData_T	In	The createData for the ML VLAN to create.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	The ML VLAN object created by CTM.

CTM support for MLVLANCreateData_T structure is as follows:

- int mlVlanId

  Supported. The ML VLAN ID used to create the ML VLAN. If it is –1, CTM assigns the ID for the ML VLAN created. CTM supports ML VLAN IDs from 1 to 4095.

- string userLabel

  Not supported.

- boolean forceUniqueness

  Supported. CTM always enforces the uniqueness of the ML VLAN ID.

- string owner

  Not supported.

- multilayersubnetwork::MLVLANTPDataList_T tpList

  Supported. A list of front Ethernet ports associated with the ML VLAN.

Each MLVLANTPData_T structure has the following members:

- globaldefs::NamingAttributes_T tpName

  Specifies the front Ethernet port name.

- multiLayerSubnetwork::L2PortType_T portType

  Specifies the port type to configure. CTM supports the following port types:

  – L2PortType_T.L2PT_UNI

  – L2PortType_T.L2PT_NNI

- multiLayerSubnetwork::L2ConnectionType_T connectionType

  Specifies the connection type to configure for the specified port type. If the port type is L2PortType_T.UNI, the following connection types are supported:

  – L2ConnectionType_T.L2CT_QINQ

  – L2ConnectionType_T.L2CT_DOT1Q

  – L2ConnectionType_T.L2CT_UNTAGGED

  If the port type is L2PortType_T.L2PT_NNI, the following connection type is supported:

  – L2ConnectionType_T.L2CT_DOT1Q

  If the connection type is not L2ConnectionType_T.L2CT_QINQ, you can specify the following RSTP values.

- multiLayerSubnetwork::L2RSTPState_T rstpState

  The NMS can enable or disable the RSTP state for a non-QinQ drop. CTM supports two port states:

  – L2RSTPState_T.L2RSTP_ENABLE

  – L2RSTPState_T.L2RSTP_DISABLE

  For a QinQ drop, this value (if specified) is ignored.

- multiLayerSubnetwork::L2PortState_T portState

  NMS can enable a service or disable a port. CTM support two port states:

  – L2PortState_T.L2PS_ENABLE

  – L2PortState_T.L2PS_DISABLE

- multiLayerSubnetwork::L2InterfaceType_T interfaceType

  Specifies the interface type to be configured. CTM supports the following interface types:

  - L2InterfaceType_T. L2IT_FASTETHER
  - L2InterfaceType _T. L2IT_GIGAETHER
  - L2InterfaceType _T. L2IT_PORTCHANNEL

- multiLayerSubnetwork::QOSPolicyData_T qosPolicyData

  CTM support for QOSPolicyData_T structure is described in QOSPolicyData_T.

- int portVlan

  The user VLAN ID to associate with this ML VLAN. If the connection type is L2ConnectionType_T.L2CT_DOT1Q, the valid port VLAN range is from 2 to 4096. If the connection type is L2ConnectionType_T.L2CT_UNTAGGED, the port VLAN is ignored.

- globaldefs::NVSList_T additionalInfo

  Supported. You can specify the customerID and service ID.

CTM support for MLVLAN_T structure is as follows:

- globaldefs::NamingAttributes_T name

  Supported. This field contains three tuples. The first tuple has the EMS value. The second tuple has the L2 topology name. The third tuple has the ML VLAN ID that is the same as nativeEMSName.

- string userLabel

  Not supported.

- string nativeEMSName

  Supported. The ML VLAN ID is reported.

- string owner

  Not supported.

- multilayersubnetwork::MLVLANTPDataList_T tpList

  Supported. Specifies the list of all front Ethernet ports and associated service type.

  If the port type on a port cannot be determined, CTM represents the port type as L2PortType_T.L2PT_UNKNOWN.

  If the connection type for an ML VLAN on a port cannot be determined, CTM represents the connection type as L2ConnectionType_T.L2CT_UNKNOWN.

- multiLayerSubnetwork::MLVLANState_T vlanState

  Supported. Valid values are MLVLANState_T.COMPLETE and MLVLANState_T.INCOMPLETE. A state change event is generated if the VLAN state changes.

- globaldefs::NVSList_T additionalInfo

  Supported. The customer ID and service ID are reported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when one of the following conditions occurs:

- l2TopologyName does not reference an L2 topology object.
```

```
- MLVLAN name does not follow the naming convention.
- QOSPolicy is specified incorrectly.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be created due to the restriction
that an L2 topology can have a maximum of 4095 ML VLAN IDs.
```

Compliance

CTM-defined.

3.10.2 modifyMLVLANAddFrontPorts

Synopsis

```
void modifyMLVLANAddFrontPorts(in globaldefs::NamingAttributes_T mlVlanName,
        in multiLayerSubnetwork::MLVLANTPDataList_T tpList,
        out multiLayerSubnetwork::MLVLAN_T mlVlan, String protocol)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface adds additional front ports to the existing MVLAN. An attribute change event is reported when the front ports are successfully added. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain updated MLVLAN_T structure information. If the addition of any of the front ports to the ML VLAN fails, CTM generates a processing failure exception. The NMS user must requery the ML VLAN information to obtain the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which to modify the ML VLAN.
tpList	multiLayerSubnetwork::MLVLANTPData List_T	In	List of front ports to include for the specified ML VLAN.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN object returned after modification.
protocol	String	In	Bridge group protocol of the ML VLAN.

For details about the MLVLAN_T, MLVLANTPData_T structure, see 3.10.1 createMLVLAN, page 3-180.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object
or when one or more PTPs in the PTP list does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist or when one or more PTPs in the PTP list does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be modified.
```

Compliance

CTM-defined.

3.10.3 modifyMLVLANDropFrontPorts

Synopsis

```
void modifyMLVLANDropFrontPorts(
        in globaldefs::NamingAttributes_T mlVlanName,
        in globaldefs::NamingAttributesList_T tpList,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface drops front ports associated with an existing ML VLAN. An attribute change event is reported when the front ports are dropped from the ML VLAN. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain updated MLVLAN_T structure information. If removal of any front ports associated with the ML VLAN fails, CTM generates a processing failure exception. The NMS user must requery the ML VLAN information to obtain the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which this ML VLAN is created.
tpList	globaldefs::NamingAttributesList_T	In	List of front PTP names to drop from the specified ML VLAN.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN object returned after modification.

For details about the MLVLAN_T structure, see 3.10.1 createMLVLAN, page 3-180.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object
or when one or more PTPs in the PTP list does not reference a PTP object.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist or when one or more PTPs in the PTP list does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be modified.
```

Compliance

CTM-defined.

3.10.4 modifyMLVLANPortState

Synopsis

```
void modifyMLVLANPortState(
        in globaldefs::NamingAttributes_T mlVlanName,
        in multiLayerSubnetwork::MLVLANPortStateList_T tpPortStateList,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables or disables specified ML VLAN traffic on the specified list of front ports. An attribute change event is reported when one of the ML VLAN parameters is modified. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain updated MLVLAN_T structure information. If modification of any of the ML VLAN port states fails, CTM generates a processing failure exception. The NMS user must requery the ML VLAN information to obtain the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which to create the ML VLAN.
tpPortStateList	multiLayerSubnetwork::MLVLANTP PortStateList_T	In	List of front ports on which to modify the port state for the specified ML VLAN.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN object returned after modification.

For details about the MLVLAN_T structure, see 3.10.1 createMLVLAN, page 3-180.

CTM support for MLVLANTPPortState_T structure is as follows:

- globaldefs::NamingAttributes_T tpName

  The front Ethernet port name used by this ML VLAN.

- L2PortState_T portState

  The portState is applicable only to the specified ML VLAN. The supported states are:

  - L2PortState_T.L2PS_ENABLE
  - L2PortState_T.L2PS_DISABLE

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object
or when one or more PTPs in the PTP list does not reference a PTP object.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist or when one or more PTPs in the PTP list does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be modified.
```

Compliance

CTM-defined.

3.10.5 modifyMLVLANRSTPState

Synopsis

```
void modifyMLVLANRSTPState(
        in globaldefs::NamingAttributes_T mlVlanName,
        in multiLayerSubnetwork::MLVLANTPrstpStateList_T tpRstpStateList,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables or disables specific ML VLAN traffic on the specified list of drop ports. An attribute change event is reported when an ML VLAN parameter is modified. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain updated MLVLAN_T structure information. If the modification of an ML VLAN RSTP state fails, CTM generates a processing failure exception. You must issue another query for the ML VLAN information to obtain the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which to create the ML VLAN.
tpRstpStateList	multiLayerSubnetwork:: MLVLANTPrstpStateList_T	In	List of front ports on which the RSTP state must be modified for the specified ML VLAN.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN object returned after modification.

For details about the MLVLAN_T structure, see 3.10.1 createMLVLAN, page 3-180.

CTM support for MLVLANRSTPState_T structure is as follows:

- globaldefs::NamingAttributes_T tpName

  The front Ethernet port name used by this ML VLAN.

- L2RSTPState_T rstpState

  The rstpState is applicable only to the specified non-QinQ drops of the ML VLAN. The supported states are:

  – L2RSTPState_T.L2RSTP_ENABLE

  – L2RSTPState_T.L2RSTP_DISABLE

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object
or when one or more PTPs in the PTP list does not reference a PTP object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist or when one or more PTPs in the PTP list does not exist.

EXCPT_QINQ - Raised when the drop port for which the RSTP state must be enabled or
disabled uses the QinQ connection type.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be modified.
```

Compliance

CTM-defined.

3.10.6 modifyMLVLANTPQOS

Synopsis

```
void modifyMLVLANTPQOS(
        in globaldefs::NamingAttributes_T mlVlanName,
        in multiLayerSubnetwork::MLVLANTPQOSPolicyList_T tpQosPolicyList,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface modifies the QoS policies of a list of termination points in the specified ML VLAN. An attribute change event is reported for each call to this API, even if more than one TP QoS policy has been modified. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain updated MLVLAN_T structure. Modification of any QoS policy is done on a best effort basis. If modification of any part of the QOSPolicy fails, CTM generates a processing failure exception. The NMS user must requery the ML VLAN information to get the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which this ML VLAN is created.
tpqosPolicyList	multiLayerSubnetwork::MVLANTPQOSPolicyList_T	In	New set of QOSPolicy_T to set on the specified ML VLAN TPs.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN TP object returned after modification.

For details about the MLVLAN_T and QOSPolicy_T structures, see 3.10.1 createMLVLAN, page 3-180.

CTM support for MLVLANTPQOSPolicy_T structure is as follows:

- globaldefs::NamingAttributes_T tpName
  - The front Ethernet port name used by this ML VLAN to modify the QoS policy.
- QOSPolicyData_T qosPolicyData
  - Refer to QOSPolicyData_T.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when one of the following conditions occurs:

- The mlVlanName passed does not have the correct format for ML VLAN name.
- The tpQosPolicyList contains the incorrect format for TP or QOS policy.
- The tpQosPolicyList contains the same TP more than once.

EXCPT_ENTITY_NOT_FOUND - Raised when one of the following conditions occurs:

- The mlVlanName passed does not reference an ML VLAN object.
- The tpQosPolicyList contains TPs that do not reference valid objects.

EXCPT_UNABLE_TO_COMPLY - Raised when the QOS policy cannot be modified.
```

Compliance

CTM-defined.

3.10.7 modifyMLVLANCustomerIDServiceID

Synopsis

```
void modifyMLVLANCustomerIDServiceID(
        in globaldefs::NamingAttributes_T mlVlanName,
        in multiLayerSubnetwork::NVSList_T ids,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface modifies the customer ID and/or service ID associated with an existing ML VLAN. An attribute change event is reported when the customer ID or service ID is modified. CTM does not specify which attribute of the ML VLAN has changed. The NMS must obtain an updated MLVLAN_T structure information. If modification of the customer ID or service ID fails, CTM generates a processing failure exception. The NMS user must requery the ML VLAN information to get the new set of parameters associated with the ML VLAN.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which this ML VLAN is created.
ids	globaldefs::NVSList_T	In	Name and value string. The name string is CustomerID and/or ServiceID, and the value string is the new customer ID or service ID.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	ML VLAN object returned after modification.

For details about the MLVLAN_T structure, see 3.10.1 createMLVLAN, page 3-180.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the ML VLAN cannot be modified.
```

Compliance

CTM-defined.

3.10.8 getMLVLAN

Synopsis

```
void getMLVLAN(
        in globaldefs::NamingAttributes_T mlVlanName,
        out multiLayerSubnetwork::MLVLAN_T mlVlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface reports the complete ML VLAN information for a specified ML VLAN name.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name with which to obtain the ML VLAN.
mlVlan	MultiLayerSubnetwork::MLVLAN_T	Out	MLVLAN_T structure.

For details about the MLVLAN_T structure, see 3.10.1 createMLVLAN, page 3-180.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when the ML VLAN name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when the ML VLAN name does not exist on the L2 topology.
```

Compliance

CTM-defined.

3.10.9 deleteMLVLAN

Synopsis

```
void deleteMLVLAN(
        in globaldefs::NamingAttributes_T mlVlanName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface deletes an existing ML VLAN. CTM generates a processing failure exception if the ML VLAN deletion fails. CTM generates an object deletion event when the ML VLAN is successfully deleted.

Parameters

Name	Type	Input/Output	Description
mlVlanName	mlVlan:MLVLAN_T	In	ML VLAN name to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when mlVlanName does not reference an ML VLAN object.

EXCPT_ENTITY_NOT_FOUND - Raised when mlVlanName references an ML VLAN object that does not
exist.

EXCPT_UNABLE_TO_COMPLY – Raised when mlVlan deletion fails.
```

Compliance

CTM-defined.

3.10.10 getAllMLVLANs

Synopsis

```
void getAllMLVLANs(
        in globaldefs::NamingAttributes_T l2TopologyName,
        in unsigned long how_many,
        out multiLayerSubnetwork::MLVLANList_T mlVlanList,
        out multiLayerSubnetwork::MLVLANIterator_I mlVlanIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists all ML VLANs for the L2 topology. The NMS user must specify the L2 topology name. CTM returns a list of all ML VLANs associated with the L2 topology specified.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	L2 topology name for which to obtain all the ML VLANs.
how_many	unsigned long	In	Maximum number of ML VLANs to return in the first batch.
mlVlanList	MLVLANList_T	Out	List of available ML VLAN names in the L2 topology.
mlVlanIt	MultiLayerSubnetwork::MLVLANIterator_I	Out	Iterator used to retrieve the remaining ML VLANs.

For details about the MLVLAN_T structure, see 3.10 L2 Topology and ML VLAN for ML-Series Ethernet Cards, page 3-180.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.10.11 getAllMLVLANNames

Synopsis

```
void getAllMLVLANNames(
        in globaldefs::NamingAttributes_T l2TopologyName,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface lists all the ML VLAN names for the L2 topology.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	L2 topology name for which to obtain all the ML VLANs.
how_many	unsigned long	In	Maximum number of ML VLANs to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	List of available ML VLAN names in this L2 topology.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator used to retrieve the remaining ML VLAN names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.
```

```
EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.10.12 getAllAssociatedSNCs

Synopsis

```
void getAllAssociatedSNCs(
        in globaldefs::NamingAttributes_T l2TopologyName,
        in unsigned long how_many,
        out subnetworkConnection::SubnetworkConnectionList_T sncList,
        out subnetworkConnection::SNCIterator_I sncIt,
        out subnetworkConnection::VCATList_T vcatList,
        out subnetworkConnection::VCATIterator_I vcatIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows CTM to report all the underlying Layer 1 SONET/SDH SNCs that are being used by the specified L2 topology.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	L2 topology name for which to obtain all the SNCs.
how_many	unsigned long	In	Maximum number of SNCs to return in the first batch.
sncList	subnetworkConnection::SubnetworkConnectionList_T	Out	First batch of SNCs.
sncIt	subnetworkConnection::SNCIterator_I	Out	Iterator to retrieve the remaining SNCs.
vcatList	subnetworkConnection::VCATList_T	Out	First batch of VCAT SNCs.
vcatIt	subnetworkConnection::VCATIterator_I	Out	Iterator to retrieve the remaining VCAT SNCs.

For details about the SubnetworkConnection_T structure, see 3.8.2 getAllSNCs, page 3-83.

For details about the VCAT_T structure, see 3.9.21 getVCATSNC, page 3-150.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.10.13 getL2Topology

Synopsis

```
void getL2Topology(
        in globaldefs::NamingAttributes_T l2TopologyName,
        out multiLayerSubnetwork::L2Topology_T l2Topo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the L2 topology structure for a given L2 topology name.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	Name of the L2 topology to retrieve.
l2Topo	multiLayerSubnetwork::L2Topology_T	Out	Retrieved L2 topology.

For details about the L2Topology_T structure, see 3.3.12 getEMS, page 3-33.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an L2 topology object that
does not exist.
```

Compliance

CTM-defined.

3.10.14 getAssociatedL2TopologyForSNC

Synopsis

```
void getAssociatedL2TopologyForSNC(
        in globaldefs::NamingAttributes_T sncName,
        out multiLayerSubnetwork::L2Topology_T l2Topo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the L2 topology structure associated with a given SNC name. CTM reports the L2 topology using the specified Layer 1 SNC. CTM returns an empty L2Topology_T structure if the SNC name specified is currently not associated with any L2 topology in the EMS.

Parameters

Name	Type	Input/Output	Description
sncName	globaldefs::NamingAttributes_T	In	Name of the SNC to retrieve the associated L2 topology.
l2Topo	multiLayerSubnetwork::L2Topology_T	Out	Retrieved L2 topology.

For details about the L2Topology_T structure, see 3.3.13 getAllL2Topologies, page 3-35.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when sncName does not reference a subnetwork connection
object.

EXCPT_ENTITY_NOT_FOUND - Raised when sncName references an SNC object that does not exist.
```

Compliance

CTM-defined.

3.10.15 getAssociatedL2TopologyForMLVLAN

Synopsis

```
void getAssociatedL2TopologyForMLVLAN(
        in globaldefs::NamingAttributes_T mlVlanName,
        out multiLayerSubnetwork::L2Topology_T l2Topo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the L2 topology structure associated with a given ML VLAN name. CTM returns an empty L2Topology_T structure if the ML VLAN name specified is currently not associated with any L2 topology in the EMS.

Parameters

Name	Type	Input/Output	Description
mlVlanName	globaldefs::NamingAttributes_T	In	Name of the ML VLAN to retrieve containing the L2 topology.
l2Topo	multiLayerSubnetwork::L2Topology_T	Out	Retrieved L2 topology.

For details about the L2Topology_T structure, see 3.3.13 getAllL2Topologies, page 3-35.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when mlVlanName does not reference an ML VLAN object.
```

```
EXCPT_ENTITY_NOT_FOUND - Raised when mlVlanName references an ML VLAN object that does not
exist.
```

Compliance

CTM-defined.

3.10.16 createL2Topology

Synopsis

```
void createL2Topology(
        in multiLayerSubnetwork::L2TopoCreateData_T createData,
        out multiLayerSubnetwork::L2Topology_T l2Topo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a way to create an L2 topology, including all the optical SNCs associated with it. The user must associate ML VLANs using the interface method defined in 3.10.1 createMLVLAN, page 3-180. CTM creates an L2 topology within the same MultiLayerSubnetwork. If the NMS specifies the L2 topology createData with MEs belonging to a different MultiLayerSubnetwork, CTM returns the ProcessingFailureException EXCPT_UNABLE_TO_COMPLY.

CTM supports point-to-point, hub-and-spoke, Resilient Packet Ring (RPR), and 802.17 RPR topologies. However, CTM supports the creation of only RPR, 802.17 RPR, and point-to-point topologies. CTM treats a hub-and-spoke topology as multiple point-to-point topologies.

This is an asynchronous operation. CTM returns this method after all requests for creating the underlying SNCs are complete. CTM autodiscovers an L2 topology based on the L1 SNCs created. CTM generates OCE for all the L1 SNCs created. CTM also generates L2 topology OCE after the L2 Topology is autodiscovered.

To create an RPR or 802.17 RPR topology, you must specify the complete list of ML-series and/or ML2 (L2/L3) equipment names to involve in the topology. RPR consists of all ML-series equipment or all ML2 (L2/L3) equipment or a combination of ML and ML2 (L2/L3) equipment. To create a point-to-point topology, you must specify the end TPs of the underlying L1 SNC.

During the L2 topology creation, if any part of the underlying L1 SNC creation fails, CTM generates a processing failure exception. However, CTM autodiscovers the L2 topology in INCOMPLETE state if any SNC was created before the failure occurred. Either the NMS creates the remaining L1 SNCs that failed, using the interface described in 3.9.16 createAndActivateSNCFromUserLabel, page 3-138 to complete the L2 topology, or the NMS deletes the L2 topology and recreates it using this operation.

For RPR topologies, the NMS must specify all the equipment to involve, in cyclical order and in pairs. There are three different scenarios:

- All SNCs in the RPR are CCAT. In this case, only SNCCreateDataList_T is populated and VCATSNCCreateDataList_T is empty. For example, if the NMS is trying to create an RPR topology involving three ML-series cards (M1, M2, and M3), the NMS must specify a three-element array of SNCCreateData_T. The first element's aEnd will specify the M1 equipment name and the zEnd will specify the M2 equipment name. The second element's aEnd will specify the M2 equipment name and zEnd will specify the M3 equipment name. The third element's aEnd will specify the M3 equipment name and zEnd will specify the M1 equipment name.
- All SNCs in the RPR are VCAT. In this case, only VCATSNCCreateDataList_T is populated and SNCCreateDataList_T is empty. For example, if the NMS is trying to create an RPR topology involving three ML-series cards (M1, M2, and M3), the NMS must specify a three-element array of VCATSNCCreateData_T. The first element's aEnd will specify the M1 equipment name and zEnd

will specify the M2 equipment name. The second element's aEnd will specify the M2 equipment name and zEnd will specify the M3 equipment name. The third element's aEnd will specify the M3 equipment name and zEnd will specify the M1 equipment name.

- Mix of CCAT and VCAT SNCs in the RPR. In this case, both VCATSNCCreateDataList_T and SNCCreateDataList_T are populated and all the equipment in the two arrays is in cyclical order.

  For example, if the NMS is trying to create an RPR topology involving three ML-series cards (M1, M2, and M3) and the SNC between M1-M2 and M3-M1 is of type CCAT and M2-M3 is of type VCAT, the NMS must specify:

  - A two-element array of SNCCreateData_T with the first element's aEnd specifying the M1 equipment name and the zEnd specifying the M2 equipment name, and the second element's aEnd specifying the M3 equipment name and zEnd specifying the M1 equipment name.
  - A one-element array of VCATSNCCreateData_T with the first element's aEnd specifying the M2 equipment name and zEnd specifying the M3 equipment name.

For point-to-point topologies, the NMS specifies either CCATSNCCreateData_T or VCATSNCCreatedata_T. For creating CCAT point-to-point topologies, CCATSNCCreateData_T is populated and VCATSNCCreateData_T is empty. For creating VCAT point-to-point topologies, VCATSNCCreateData_T is populated and CCATSNCCreateData_T is empty.

In addition to specifying SNC creation parameters using the SNCCreateData_T structure for each SNC in the L2 topology, the NMS can provide a partial or complete list of managed elements, or of both end PTPs of a topological link that is included or excluded in the SNC route during SNC creation. The NMS can specify only one type of object name list during the SNC creation. CTM returns a processing failure exception if the NMS-specified list is not homogeneous. CTM returns a processing failure exception with EXCPT_ENTITY_NOT_FOUND as the error reason if any one of the specified managed elements or PTPs does not exist. If the NMS specifies a PTP list, CTM verifies whether the PTPs are specified in pairs and each PTP in a pair is associated with each end PTP of the same topological link. CTM uses the first PTP of a pair as the aEnd of the topological link and uses the second PTP of the same pair as the zEnd of the topological link. CTM attempts to route the SNC from aEnd PTP of a PTP pair to zEnd PTP of the same pair on a topological link. CTM returns a processing failure exception if the SNC cannot be routed from the specified aEnd PTP to the zEnd PTP of a topological link. The NMS then must interchange the aEnd and zEnd PTPs specified, provide a different PTP pair, or delete the PTP pair from the include or exclude list. CTM returns a processing failure exception if at least one PTP is not associated with a topological link. The NMS must ensure that the sequence of the object name list conforms to the sequence of the objects existing in the network. CTM returns a processing failure exception if the sequence is nonconforming.

CTM does not support the creation of L2 topologies in which SNCs have more than one source point and destination endpoint in SNCCreateData_T.aEnd or SNCCreateData_T.zEnd.

CTM does not support 802.17 RPR topology creation if the circuit size entered is not compatible with the bandwidth values entered. CTM returns a processing failure exception if the circuit size and bandwidth values are incompatible. Refer to the *Cisco Transport Manager Release 9.2 User Guide* for compatibility between circuit size and bandwidth.

CTM supports the creation of L2 topologies containing VCAT SNCs. For point-to-point topologies, either the CCAT SNC or VCAT SNC is specified. RPR topologies can consist of both CCAT and VCAT SNCs.

Parameters

Name	Type	Input/Output	Description
L2TopoCreateData_T	multiLayerSubnetwork::L2TopoCreateData_T	In	Data structure defining the L2 topology to create.
L2Topo	multiLayerSubnetwork::L2Topology_T	Out	Created L2 topology.

CTM support for L2TopologyCreateData_T structure is as follows:

- string nativeEMSName

  Supported. Used as the L2 topology name.

- string userLabel

  Not supported

- boolean forceUniqueness

  Guarantees the uniqueness if all L2 topologies are created by CTM only. If any other tool is used for L2 topology creation, CTM does not guarantee uniqueness.

- string owner

  Not supported.

- transmissionParameters::LayerRate_T

  Not supported. LayerRate is specified as part of SNCCreateData_T an and VCAT SNCCreateData_T structure

- multiLayerSubnetwork::L2TopoType l2TopoType

  The type of the L2 topology. Valid values are L2TopoType.PT2PT, L2TopoType.RPR, and L2TopoType.DOT17_RPR.

- subnetworkConnection::SNCCreateDataList_T sncInfoList

  Supported.

  CTM support for SNCCreateData_T structure for the L2 topology creation is as follows:

  - string userLabel

    Not supported.

  - boolean forceUniqueness

    Not supported.

  - string owner

    Not supported.

  - globaldefs::ConnectionDirection_T direction

    Not supported. CTM always creates CD_BI (bidirectional) circuit.

  - StaticProtectionLevel_T staticProtectionLevel

    Supported. CTM supports UNPROTECTED and FULLY_PROTECTED.

  - ProtectionEffort_T protectionEffort

    CTM supports all values defined by TMF.

    If the NMS requests unprotected SNC on a BLSR ring, CTM creates the SNC. The same is true for unprotected SNC on 1+1 protection.

If the NMS specifies EFFORT_SAME_OR_WORSE or EFFORT_WHATEVER, CTM expects the value of staticProtectionLevel to be UNPROTECTED. These values indicate that CTM must create the PCA.

  - Reroute_T rerouteAllowed

    Not supported.

  - NetworkRouted_T networkRouted

    Not supported.

  - SNCType_T sncType

    Not supported.

  - transmissionParameters::LayerRate_T layerRate

    Supported. For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- CrossConnectList_T ccInclusions

  Not supported.

- ResourceList_T neTpInclusions

  Supported. You must provide a list of managed element names or the PTP names at the endpoints of the topological link. For complete manual routing, you must provide the list of all CTPs of the links (OCH trail links, if any) to which the circuit must be routed.

- boolean fullRoute

  - Not supported.

- ResourceList_T neTpSncExclusions

  Supported. CTM expects a list of ME or PTP names at the endpoints of the topological link to exclude from the SNC route.

- globaldefs::NamingAttributesList_T aEnd

  Supported. If the topology is RPR, you specify an ML-series equipment name to use to create the L2 topology. If the topology is point-to-point, you specify the aEnd TP name. Point-to-point topologies can be created with one endpoint on the ML-series equipment and the other endpoint on ML-series, G-series, or OC-n equipment.

- globaldefs::NamingAttributesList_T zEnd

  Supported. If the topology is RPR, you specify an ML-series equipment name to use to create the L2 topology. If the topology is point-to-point, you specify the zEnd TP name. Point-to-point topologies can be created with one endpoint on the ML-series equipment and the other endpoint on ML-series, G-series, or OC-n equipment.

- globaldefs::NVSList_T additionalCreationInfo

  Supported. This list can be used to provide additional parameters for SNC creation. CTM supports the following values.

Name	Value	Description
upsrRevertiveMode	Revertive or nonrevertive	Supported only for point-to-point L2 topology. If value is set to revertive, traffic reverts back to the working path when the conditions that diverted it to the protect path are repaired. Otherwise it remains on the protect path after the switch.
upsrRevertiveTime	Valid float value from 0.5 to 12, in increments of 12	Supported only for point-to-point L2 topology. If SNC is revertive, set this parameter. This is the amount of time that will elapse before the traffic reverts to the working path. The default value is 5 minutes. **Note** CTM does not validate any of these values.
upsrSfBerLevel	3, 4, or 5	Supported only for point-to-point L2 topology. Set the thresholds for path protection, path-level signal failure, and bit error rate (only for STS SNCs). **Note** CTM does not validate any of these values.
upsrSdBerLevel	5, 6, 7, 8, and 9	Supported only for point-to-point L2 topology. Set the thresholds for path protection, path-level signal degrade, and bit error rate (only for STS SNCs). **Note** CTM does not validate any of these values.
upserSwitchOnPdiP	True or false	Supported only for point-to-point L2 topology. Set this to "true" if you want traffic to switch when an STS payload defect indicator is received (only for STS SNCs).
goAndReturnRouting	True or false	Not supported.
sncDescription	String	Not supported.
VLAN	VLAN ID	Not supported.
CustomerID	String	Not supported.
ServiceID	String	Not supported.
createTunnelOnTransitNodes	True or false	Applicable for VC SNCs with one or more transit nodes that do not have a tunnel or the existing tunnel is exhausted. This parameter specifies whether to create a tunnel on the transit nodes. For SNCs without transit nodes or with an existing tunnel on a transit node that has available bandwidth, this value is ignored. If the value specified is true, a tunnel is created and the SNC is routed using this tunnel. If the value specified is false, a new tunnel is not created. In this case, if a tunnel that is not full is available on the transit node, the circuit is routed through the tunnel; otherwise, the circuit is not routed through the tunnel.

Name	Value	Description
createVTAggregation	True or false	Applicable for VC SNCs with one or more transit nodes that do not have a VAP circuit or the existing VAP circuit is exhausted. This parameter specifies whether to create a new VAP circuit on the transit node(s). If the value specified is true, a VAP circuit is created and SNC is routed using this new VAP circuit. If the value specified is false, a new VAP circuit is not created.
sourceIsSTSGrroming	True or false	When set to true, the source drop point of the VC circuit is STS groomed for the VAP circuit. **Note** This attribute is valid only when createVTAggregation is set to true. **Note** sourceIsSTSGrooming and destinationIsSTSGrooming cannot both be set to true.
destinationIsSTSGrooming	True or false	When set to true, the destination drop point of the VT or VC circuit is STS groomed for the VAP circuit. **Note** This attribute is valid only when createVTAggregation is set to true. **Note** sourceIsSTSGrooming and destinationIsSTSGrooming cannot both be set to true.
vapDestination	String	User-specified string (ME name) representing the VAP destination drop point.
usePortlessTransmux	True or false	When set to true, the portless transmux card is used for the circuit. The NMS provides information for portlessTransmuxMeName, portlessTransmuxSlot, and isSourceDS3MappedSTS; otherwise, an INVALID INPUT exception is thrown. When set to false, the portless transmux card is not used as the intermediate point for the end-to-end STS-1 circuit.
portlessTransmuxMeName	String	Not supported.
portlessTransmuxSlot	/rack=1/shelf=1/slot=[1..6][12..17]	Not supported.
isSourceDS3MappedSTS	True or false	Not supported.
vcatMemberId	Integer from 1 to 256	Used for VCAT SNC creation, to identify every VCAT member L1 SNC uniquely. For normal CCAT circuits, this value is ignored if specified.

- subnetworkConnection::VCATSNCCreateDataList_T vcatSncInfoList

  Supported.

CTM support for VCATSNCCreateData_T structure for the L2 topology creation is as follows:

- string userLabel

  Not supported.

- boolean forceUniqueness

  CTM guarantees the uniqueness of all VCATs created in CTM. It is not guaranteed if other tools were used for VCAT SNC creation.

- string owner

  Not supported.

- globaldefs::ConnectionDirection_T direction

  Not supported. CTM supports CD_BI (bidirectional).

- boolean isSymmetrical

  Supported. Only symmetric VCAT SNCs are supported. If set to true, all members are bidirectional L1 SNCs. The VCAT layer rate is the same in both directions. If set to false, all members are unidirectional L1 SNCs.

- subnetworkConnection::LCASMode_T mode

  Supported. CTM supports the following values:

  - LCASMode_T.MODE_NONE
  - LCASMode_T.MODE_SWLCAS
  - LCASMode_T.MODE_LCAS

- subnetworkConnection::RoutingMode_T routingMode

  Supported. CTM supports the following values:

  - RoutingMode_T.COMMONFIBER_ROUTED
  - RoutingMode_T.INDEPENDENTFIBER_ROUTED

- globaldefs::NamingAttributes_T aEnd;

  Supported. If the topology is an RPR topology, you specify an ML-series equipment name to use to create the L2 topology. If the topology is a point-to-point topology, CTM expects this field to be an FTP representing the information up to the slot level of the card involved in the VCAT SNC.

- globaldefs::NamingAttributes_T zEnd;

  Supported. If the topology is an RPR topology, you specify an ML-series equipment name to use to create the L2 topology. If the topology is a point-to-point topology, CTM expects this field to be an FTP representing the information up to the slot level of the card involved in the VCAT SNC.

- subnetworkConnection::SNCCreateDataList_T sncCreateDataList

  Supported. At least one L1 SNC is created at the time of VCAT SNC creation. CTM expects you to specify the create data information required to create the L1 SNCs in the data structure.

- globaldefs::NVSList_T additionalCreationInfo

  Not supported.

- subnetworkConnection::StaticProtectionLevel_T staticProtectionLevel

  Not supported. StaticProtectionLevel is specified as part of SNCCreateData_T and VCATSNCCreateData_T structure.

- globaldefs::NVSList_T additionalInfo

Supported. You can choose to specify bandwidth parameters. By default, bandwidth parameters have a zero value. If you choose to enter a specific bandwidth value, you must also specify its corresponding CoS value. The default value for all CoS values is N/A. The sum of all the bandwidth parameters must be from 1 to 99. The CoS commit value must be from 0 to 7. The following are the supported bandwidth parameters (Group1Bandwidth and Group2Bandwidth are multicast bandwidth parameters, and their corresponding CoS values are Group1CoS and Group2CoS):

- Group1Bandwidth
- Group2Bandwidth
- SpMgmtBandwidth
- CommittedBandwidth
- BestEffortBandwidth
- AvvidControlBandwidth
- ClassSpMgmt
- ClassCommitted
- ClassBestEffort
- ClassAvvidControl
- ClassAvvidVoiceVideo
- ClassAReservedBandwidth
- ClassA1Bandwidth
- ClassBCommittedBandwidth
- SpMgmtCoS
- CommittedCoS
- AvvidVoiceVideoCoS
- AvvidControlCoS
- CosCommit
- Group1CoS
- Group2CoS

Caution Group2Bandwidth parameters must be entered along with Group1Bandwidth parameters. Entering Group2Bandwidth alone creates inconsistency in the L2 topology table.

If the topology is DOT17_RPR, multicast bandwidth parameters are not supported; Group1Bandwidth and Group2Bandwidth values are not needed. Group1CoS and Group2CoS do not apply to DOT17_RPR. For DOT17_RPR topologies, you can specify the absolute bandwidth values for the three possible classes of traffic (A Reserved, A1, and B Committed). For the remaining bandwidth parameters, you specify to which class of traffic (Class A, B, or C) the parameter belongs. For example, ClassSpMgmt belongs to Class C-type traffic.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_INVALID_INPUT - Raised when the input tpList is invalid or when the specified layer
rate is an invalid layer rate. This exception is also raised when the user does not
specify the corresponding CoS values for the bandwidth parameters specified.

EXCPT_ENTITY_NOT_FOUND - Raised when the input tpList does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when a layer rate is not supported for L2 topology
creation or when CTM cannot create the L2 topology for the specified layer rate. Raised if
the VCAT SNC cannot be created and activated because the CTP conflicts with another VCAT
SNC or the NE does not support the VCAT SNC creation.

EXCPT_OBJECT_IN_USE - Raised when the specified L2 topology already exists.

EXCPT_NE_COMM_LOSS - Raised when one of the managed elements specified as part of the L2
topology creation is not reachable.
```

Compliance

CTM-defined.

3.10.17 deleteL2Topology

Synopsis

```
void deleteL2Topology(
        in globaldefs::NamingAttributes_T l2TopologyName)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a method to delete an L2 topology and all the SNCs associated with it. If there are any ML VLANs associated with the L2 topology when this interface method is invoked, CTM generates a processing failure exception. The NMS must ensure that there are no ML VLANs associated with the L2 topology, to ensure that the L2 topology is successfully deleted.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	Name of the L2 topology to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an object that does not
exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the l2TopologyName specified has an ML VLAN
associated with it.

EXCPT_UNABLE_TO_COMPLY – Raised when one or more of the managed elements associated with
the specified L2 topology are not accessible.
```

Compliance

CTM-defined.

3.10.18 modifyL2Topology

Synopsis

```
void modifyL2Topology(
        in globaldefs::NamingAttributes_T l2TopologyName,
        in globaldefs::NVList_T l2ModifyData,
        out multiLayerSubnetwork::L2Topology_T l2Topo)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a way to modify an L2 topology. It enables the OSS to modify the L2 topology for the bandwidth allocation parameters. For more information on the bandwidth allocation and CoS commit parameters, see 3.10.16 createL2Topology, page 3-195. This interface is used when the L2 topology is discovered, but it cannot be used when the L2 topology synchronization state is L2TopoSyncState.L2NOTREADY.

Parameters

Name	Type	Input/Output	Description
l2TopologyName	globaldefs::NamingAttributes_T	In	Name of the L2 topology to modify.
l2Modifydata	globaldefs::NVList_T	In	Parameters for modification of the L2 topology; the bandwidth allocation parameters.
l2Topo	multiLayerSubnetwork::L2Topology_T	Out	Modified L2 topology.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when l2TopologyName does not reference an L2 topology object.

EXCPT_ENTITY_NOT_FOUND - Raised when l2TopologyName references an object that does not
exist.

EXCPT_UNABLE_TO_COMPLY - Raised when one or more of the managed elements associated with
the specified L2 topology are not accessible.
```

Compliance

CTM-defined.

3.10.19 getAllAvailableMLEquipmentOrTPNames

Synopsis

```
void getAllAvailableMLEquipmentOrTPNames(
        in globaldefs::NamingAttributesList_T managedElementList,
        in multiLayerSubnetwork::L2TopoType_T topoType,
        in unsigned long how_many,
        out globaldefs::NamingAttributesList_T nameList,
        out globaldefs::NamingAttributesIterator_I nameIt)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface provides a list of all available ML-series equipment names that can be used in creating an RPR L2 Topology, or a list of all available ML-series TP names that can be used in creating a point-to-point L2 topology. The returned list of ML-series equipment or TP names does not guarantee capability to create an L2 Topology. The list of ML-series equipment or TP names indicates the equipment or TP that is not participating in an L2 topology at this time.

Parameters

Name	Type	Input/Output	Description
managedElementList	globaldefs::NamingAttributesList_T	In	List of MEs from which available ML-series cards must be determined.
topoType	multiLayerSubnetwork::L2TopoType_T	In	Topology type that is created using the list. CTM supports L2TopoType_T.L2TT_RPR or L2TopoType_T.L2TT_PT2PT.
how_many	unsigned long	In	Maximum number of equipment or TP names to return in the first batch.
eqorTPNamesList	NamingAttributesList_T	Out	First batch of equipment or TP names.
eqorTPNamesIt	NamingAttributesIterator_I	Out	Iterator used to retrieve the remaining equipment or TP names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when any of the managed element names do not refer to a
managed element object.

EXCPT_ENTITY_NOT_FOUND - Raised when any of the managed elements do not exist in CTM.
```

Compliance

CTM-defined.

3.11 E-Series VLAN Interfaces

VLAN is required for single-card mode or multicard mode Ethernet SNC creation on E-series cards for ONS 15327 and ONS 15454 NEs. CTM provides methods to create and delete VLAN. In CTM, VLAN is created within a subnetwork and is available to all objects of a subnetwork in EMS. Each VLAN is described by a unique name, an ID, and the name of the multilayer subnetwork. A network partition can contain more than one multilayer subnetwork. VLAN names and IDs are unique within the network partition.

```
typedef long VlanId;
struct Vlan_T {
        VlanId id;
        string name;
        globaldefs::NamingAttributes_T subnetname;
};
```

3.11.1 createVLAN

Synopsis

```
void createVLAN in globaldefs::NamingAttributes_T subnetname,
        in VlanId vlanId,
        in string name,
        out Vlan_T vlan)
     raises(globaldefs::ProcessingFailureException);
```

Description

Creates a new VLAN with an ID and name associated with the multilayer subnetwork.

The scope of the VLAN is network partition wide. A network partition is a collection of the multilayer subnetwork.

Parameters

Name	Type	Input/Output	Description
subnetname	globaldefs::NamingAttributes_T	In	Multilayer subnetwork name to which to associate this VLAN.
vlanId	VlanId	In	ID of the VLAN to create.
name	String	In	Name of the VLAN to create.
vlan	Vlan_T	Out	VLAN object created by CTM.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when a VLAN ID or subnet name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when a subnet name does not exist in the EMS.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.11.2 deleteVLAN

Synopsis

```
void deleteVLAN(
        in globaldefs::NamingAttributes_T subnetname,
        in VlanId vlanId)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface deletes an existing VLAN with a VLAN ID associated with the multilayer subnetwork.

Parameters

Name	Type	Input/Output	Description
subnetname	globaldefs::NamingAttributes_T	In	Multilayer subnetwork name to which this VLAN is associated.
vlanId	VlanId	In	ID of the VLAN to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when a VLAN ID or subnet name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when a subnet name does not exist in the EMS.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.11.3 getAllVLANs

Synopsis

```
void getAllVLANs(
        in globaldefs::NamingAttributes_T subnetname,
        out VlanList_T vlanList)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns a list of all VLANs for the multilayer subnetwork. All the VLANs for the network partition that contain the multilayer subnetwork are reported.

Parameters

Name	Type	Input/Output	Description
subnetname	globaldefs::NamingAttributes_T	In	Multilayer subnetwork name for which to obtain all the VLANs.
vlanList	VlanList_T	Out	List of available VLANs for this EMS.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised when a subnet name is invalid.

EXCPT_ENTITY_NOT_FOUND - Raised when a subnet name does not exist in the EMS.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.12 nmsSession::NmsSession_I

The nmsSession module enables the EMS to inform the NMS of problems with notifications. When CTM fails to push an event, it notifies all connected NMSs by invoking the eventLossOccurred method on every active NmsSession_I. When the event loss period is over, CTM invokes the eventLossCleared method on NmsSession_I.

Both methods are implemented by the NMS.

NmsSession also implements all methods defined in 3.15 session::Session_I, page 3-287.

CTM requires that NmsSession implement two new methods that are used by CTM to notify NmsSession that the call to the getCTMHistoryPMData method has failed or succeeded.

3.12.1 historyPMDataCompleted

Synopsis

```
void historyPMDataCompleted(
        in string fileName)
```

Description

When getCTMHistoryPMData completes successfully, CTM invokes this method on the NMS session. At that time, NMS can retrieve the performance monitoring (PM) file from the CTM server.

Parameters

Name	Type	Input/Output	Description
fileName	string	In	Name of the file created by CTM in response to the getCTMHistoryPMData call made by the NMS. This name is the absolute path on CTM server.

Throws

None.

Compliance

CTM-defined.

3.12.2 historyPMDataFailed

Synopsis

```
void historyPMDataFailed(
        out string errorReason)
```

Description

When getCTMHistoryPMData fails, CTM invokes this method on the NMS session. Errors encountered by CTM are stored in the errorReason parameter.

Parameters

Name	Type	Input/Output	Description
errorReason	string	Out	Error string to indicate to the NMS the reason for the failure.

Throws

None.

Compliance

CTM-defined.

3.12.3 alarmLossOccurred

Synopsis

```
void alarmLossOccurred(
        in globaldefs::Time_T startTime,
          in string notificationId)
```

Description

When the EMS fails to discard an alarm, TCA, file transfer status, or protection switch notification, it notifies all connected NMSs by invoking this method on every active NmsSession_I. This interface is also invoked on any new NmsSession_I set up during the event loss period.

When the EMS invokes this method on the NmsSession_I, it sets an internal flag to indicate that it has informed the NMSs of the alarm loss. As long as this flag is set, the EMS does not invoke this method again. However, the EMS might invoke eventLossOccurred if it fails to push a different type of event.

When this method is invoked on an NmsSession_I, the NMS learns whether the EMS failed to push one or more alarms, TCAs, file transfers, or protection switch notifications. The NMS is out-of-sync with the EMS for these types of notifications. The NSM waits until the EMS calls eventLossCleared before resynchronizing alarms, TCAs, file transfers, and protection switch data with the EMS.

Parameters

Name	Type	Input/Output	Description
startTime	globaldefs::Time_T	In	Time of the first lost notification.
notificationId	string	In	ID of the first lost notification.

Throws

None.

Compliance

TMF-defined.

3.13 performance::PerformanceManagementMgr_I

This interface accesses operations that deal with performance element monitoring. A handle to an instance of this interface is gained through the getManager operation in emsSession. This interface is inherited from 3.2 common::Common_I, page 3-18.

```
interface PerformanceManagementMgr_I:common::Common_I
```

This interface also implements the getCapabilities method. Operations described in the following sections are returned to the NMS when it invokes getCapabilities.

3.13.1 disablePMData

Synopsis

```
void disablePMData(
in PMTPSelectList_T pmTPSelectList,
out PMTPSelectList_T failedTPSelectList)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface instructs the EMS to turn off PM data collection for a list of measurement points. Disabling is done on a best-effort basis. If PM could not be disabled for a subset of cases in the PMTPSelectList, a list identifying this subset is returned. PM collection involves collection of historical PM data. If this method is called between PM collection intervals, PM data for that interval is not collected. There are no incomplete collection periods.

Example

For PM collection intervals T-15min, T, T + 15min and so on, and a disablePMData call at T + 3min, the PM data is only collected up to and including interval T. T + 15min interval PM data is not collected.

If PM collection was never started for this TP, the operation is considered successful.

Parameters

Name	Type	Input/Output	Description
pmTPSelectList	PMTPSelectList_T	In	This structure contains the relevant data for the disablePMData request. This value cannot be empty.
failedTPSelectList	PMTPSelectList_T	Out	List of points that were not completely disabled. An empty list indicates that the total request was successful.

CTM provides the following support for the PMTPSelect_T structure for this method:

- globaldefs::NamingAttributes_T name

  The name of the object to which this selection applies. CTM only supports ManagedElement name in this field.

- transmissionParameters::LayerRateList_T layerRateList

  Not supported.

- PMLocationList_T pMLocationList

  Not supported.

- GranularityList_T granularityList

  Supported only for enablePMData. 15min and 24h are the only valid values. If the list is empty, both 15min and 24h are considered.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if pmTPSelectList is empty.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.13.2 enablePMData

Synopsis

```
void enablePMData(in PMTPSelectList_T pmTPSelectList,
out PMTPSelectList_T failedTPSelectList)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface instructs the EMS to enable (turn on) PM data collection for a list of measurement points. Enabling is done on a best-effort basis. If PM could not be enabled for a subset of cases in the pmTPSelectList, a list identifying this subset is returned. PM collection does not start immediately as required by the TMF. There are no incomplete collection periods. If PM collection was already on for a TP, the operation is considered successful.

Parameters

Name	Type	Input/Output	Description
pmTPSelectList	PMTPSelectList_T	In	This structure contains the relevant data for the enablePMData request. This value cannot be empty.
failedTPSelectList	PMTPSelectList_T	Out	List of points that were not enabled completely. An empty list indicates that the total request succeeded.

For details about the PMTPSelect_T structure, see 3.13.1 disablePMData, page 3-210.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if pmTPSelectList is empty.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.13.3 setTCATPParameter

Synopsis

```
void setTCATPParameter(in globaldefs::NamingAttributes_T tpName,
      inout TCAParameters_T tcaParameters)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface modifies the values of TCA thresholds on a TP-layer rate measurement point. This interface can be applied to a PTP or CTP. The NMS specifies the TP and layer rate, the granularity, and the values of one or more TCA thresholds to modify. The operation mode is best-effort. If no TCA value is set, the default value is active.

If CTM fails to set TCA parameters, an exception is generated. A successful return of this operation guarantees that all values are set. If a parameter fails, the remaining parameters are unchanged. CTM does not send events or updates when TCA parameters are changed. It also does not validate the range of PM parameter values.

The NMS always makes synchronous calls to this method.

Parameters

Name	Type	Input/Output	Description
TpName	globaldefs::NamingAttributes_T	In	Identification of the TP for which the values of one or more TCA parameters are set.
TcaParameters	TCAParameters_T	In/out	List of TCA IDs and corresponding values to apply to a specific layer rate. The settings that are actually applied are returned in this parameter.

CTM support for TCAParameters_T structure is as follows:

- transmissionParameters::LayerRate_T layerRate

  For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- Granularity_T granularity

  CTM supports 15 minute (15min) and 24 hour (24h).

- PMThresholdValueList_T tcaTypeValues

  This is a sequence of PMThresholdValue_T. It contains the PMParameterName_T pmParameterName field.

CTM supports the NE PM values listed in Table 3-5, Table 3-6, Table 3-7, Table 3-9, Table 3-10, and Table 3-11.

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_POS_100M			
LR_POS_GIGABIT			

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	
	PMP_RX_PACKETS		
	PMP_IFINDISCARDS		
	PMP_RX_TOTAL_ERRORS		
	PMP_IFOUTOCTETS		
	PMP_TX_PACKETS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSCSFRAISED		
	PMP_ETHERSTATSDROPEVENTS		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		
	PMP_HDLCRXABORTS		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_RXPKTSDROPINTERNALCONGESTION		
LR_Fast_Ethernet			
LR_Gigabit_Ethernet			
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RXPACKETS		
	PMP_IFINUCASTPKTS		
	PMP_IFINMULTICASTPKTS		
	PMP_IFINBROADCASTPKTS		
	PMP_IFINDISCARDS		
	PMP_IFINERRORS		
	PMP_IFOUTOCTETS		
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_TXPACKETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFOUTUCASTPKTS		
	PMP_IFOUTMULTICASTPKTS		
	PMP_IFOUTBROADCASTPKTS		
	PMP_DOT3STATSALIGNMENTERRORS		
	PMP_DOT3STATSFCSERRORS		
	PMP_DOT3STATSSINGLECOLLISIONFRAMES		
	PMP_DOT3STATSFRAMETOOLONG		
	PMP_ETHERSTATSUNDERSIZEPKTS		
	PMP_ETHERSTATSFRAGMENTS		
	PMP_ETHERSTATSPKTS64OCTETS		
	PMP_ETHERSTATSPKTS65TO127OCTETS		
	PMP_ETHERSTATSPKTS128TO255OCTETS		
	PMP_ETHERSTATSPKTS256TO511OCTETS		
	PMP_ETHERSTATSPKTS512TO1023OCTETS		
	PMP_ETHERSTATSPKTS1024TO1518OCTETS		
	PMP_ETHERSTATSBROADCASTPKTS		
	PMP_ETHERSTATSMULTICASTPKTS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ETHERSTATSOVERSIZEPKTS		
	PMP_ETHERSTATSJABBERS		
	PMP_ETHERSTATSOCTETS		
	PMP_ETHERSTATSCOLLISIONS		
	PMP_ETHERSTATSCRCALIGNERRORS		
	PMP_PORTDROPCOUNTS		
	PMP_ETHERSTATSCOLLISIONFRAMES		
	PMP_IFOUTDISCARDS		
	PMP_MEDIAINDSTATSOVERSIZEDROPPED		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
LR_COS_100M			
LR_COS_Gigabit			

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_INTERFACE_NAME	PML_NEAR_END_RX	15 min, 24 hr
	PMP_INTERFACE_DIRECTION		
	PMP_CLASS_OF_SERVICE_LEVEL		
	PMP_POST_POLICY_PACKETS		
	PMP_POST_POLICY_BYTES		
	PMP_DROP_PACKETS		
	PMP_DROP_BYTES		
LR_T1_and_DS1_1_5M			
Note VT1.5 data is retrieved using this layer rate. STS data retrieval is not supported.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
LR_GFP_Multirate			
PTP	PMP_GFPSTATSCSFRAISED	PML_NEAR_END_RX	15 min, 24 hr
	PMP_GFPSTATSLFDRAISED		
	PMP_GFPSTATSROUNDTRIPLATENCY		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXSBLKCRCERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T3_and_DS3_45M			
LR_STS1_and_AU3_High_Order_VC3			
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS48c_and_VC4_16c (ONS 15310 MA)			

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_FC		15 min, 24 hr
	PMP_SES		15 min, 24 hr
	PMP_ES		15 min, 24 hr
	PMP_UAS		
	PMP_PPJC_PDET	PML_NEAR_END_RX	
	PMP_NPJC_PDET		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PJDIFF		
	PMP_PJCS_PDET		
	PMP_PJCS_PGEN		
LR_Section_OC3_STS3c_and_RS_STM1 **LR_Section_OC12_STS12c_and_RS_STM4** **LR_Section_OC48_STS48c_and_RS_STM16 (ONS 15310 MA)**			
PTP	PMP_CV	PML_NEAR_END_RX	
	PMP_ES		
	PMP_SES		
	PMP_SEFS		
LR_Line_OC3_STS3_and_MS_STM1 **LR_Line_OC12_STS12_and_MS_STM4** **LR_Line_OC48_STS48_and_MS_STM16 (ONS 15310 MA)**			
PTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	
	PMP_ES		
	PMP_SES		

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_AISS_PBIT		
	PMP_CVP_PBIT		
	PMP_ESP_PBIT		
	PMP_SASP_PBIT		
	PMP_SESP_PBIT		
	PMP_UASP_PBIT		
	PMP_CVP_CPBIT	PML_NEAR_END_RX, PML_FAR_END_RX	
	PMP_ESP_CPBIT		
	PMP_SESP_CPBIT		
	PMP_UASP_CPBIT		
	PMP_SASP_CPBIT	PML_FAR_END_RX	

Table 3-5 ONS 15310 CL and ONS 15310 MA SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T1_and_DS1_1_5M			
PTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_CV_P	PML_NEAR_END_RX, PML_NEAR_END_TX	
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_UAS_P		
	PMP_AISS_P		
	PMP_FCAISS_P		
	PMP_CV_P	PML_FAR_END_RX	
	PMP_ES_P		
	PMP_ESA_P		
	PMP_ESB_P		
	PMP_SES_P		
	PMP_SEFS_P		
	PMP_CSS_P		
	PMP_UAS_P		
	PMP_ES		
	PMP_FC_P		
	PMP_ES		
LR_VT1_5_and_TU11_VC11			
CTP	Same as ONS 15327	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr

Table 3-6 ONS 15327 TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet			
PTP	PMP_DOT3STATSALIGNMENTERRORS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_DOT3STATSCARRIERSENSEERRORS		
	PMP_DOT3STATSDEFERREDTRANSMISSIONS		
	PMP_DOT3STATSEXCESSIVECOLLISIONS		
	PMP_DOT3STATSFCSERRORS		

Table 3-6 ONS 15327 TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			
PTP	PMP_DOT3STATSFRAMETOOLONG	PML_NEAR_END_RX	15 min, 24 hr
	PMP_DOT3STATSLATECOLLISIONS		
	PMP_DOT3STATSMULTIPLECOLLISIONFRAMES		
	PMP_DOT3STATSSINGLECOLLISIONFRAMES		
	PMP_DOT3STATSSQETESTERRORS		
	PMP_ETHERSTATSBROADCASTPKTS		
	PMP_ETHERSTATSCOLLISIONFRAMES		
	PMP_ETHERSTATSCOLLISIONS		
	PMP_ETHERSTATSCRCALIGNERRORS		
	PMP_ETHERSTATSFRAGMENTS		
	PMP_ETHERSTATSJABBERS		
	PMP_ETHERSTATSMULTICASTPKTS		
	PMP_ETHERSTATSOCTETS		
	PMP_ETHERSTATSOVERSIZEPKTS		
	PMP_ETHERSTATSPKTS1024TO1518OCTETS		
	PMP_ETHERSTATSPKTS128TO255OCTETS		
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			

Table 3-6 ONS 15327 TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_ETHERSTATSPKTS256TO511OCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ETHERSTATSPKTS512TO1023OCTETS		
	PMP_ETHERSTATSPKTS64OCTETS		
	PMP_ETHERSTATSPKTS65TO127OCTETS		
	PMP_ETHERSTATSUNDERSIZEPKTS		
	PMP_HDLCPKTDROPS		
	PMP_IFINBROADCASTPKTS		
	PMP_IFINDISCARDS		
	PMP_IFINERRORS		
	PMP_IFINMULTICASTPKTS		
	PMP_IFINOCTETS		
	PMP_IFINUCASTPKTS		
	PMP_IFOUTBROADCASTPKTS		
	PMP_IFOUTDISCARDS		
	PMP_IFOUTMULTICASTPKTS		
	PMP_IFOUTOCTETS		
	PMP_IFOUTUCASTPKTS		
	PMP_RXETHERUTILIZATIONSTATS		
	PMP_RXPAUSEFRAMES		
	PMP_RXPKTSDROPPEDINTERNALCONGESTION		
	PMP_RXPACKETS		
	PMP_TXETHERUTILIZATIONSTATS		
	PMP_TXPAUSEFRAMES		
	PMP_TXPKTSDROPPEDINTERNALCONGESTION		
	PMP_TXPACKETS		
LR_T1_and_DS1_1_5M			
Note VT1.5 data is retrieved using this layer rate. STS data retrieval is not supported.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		

Table 3-6 ONS 15327 TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T3_and_DS3_45M			
LR_STS1_and_AU3_High_Order_VC3			
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_FC		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_PPJ	PML_NEAR_END_RX	
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PJ_DIFF		
	PMP_PJP_SEC		
	PMP_PJN_SEC		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
LR_Section_OC3_STS3c_and_RS_STM1			
LR_Section_OC12_STS12c_and_RS_STM4			
LR_Section_OC48_STS48c_and_RS_STM16			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_SEFS		
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
PTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_FC		
	PMP_UAS		

Table 3-6 ONS 15327 TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16 (continued)			
PTP	PMP_PSC	PML_NEAR_END_RX	15 min, 24 hr
	PMP_PSD		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PSC_W		
	PMP_PSD_W		
	PMP_PSC_S		
	PMP_PSD_S		
	PMP_PSC_R		
	PMP_PSD_R		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
	PMP_PJ_DIFF		
LR_T3_and_DS3_45M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
LR_T1_and_DS1_1_5M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_CV_P	PML_NEAR_END_RX, PML_NEAR_END_TX	
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_UAS_P		
	PMP_AISS_P		

Table 3-6 ONS 15327 TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T1_and_DS1_1_5M (continued)			
PTP	PMP_CV_P	PML_FAR_END_RX	15 min, 24 hr
	PMP_ES_P		
	PMP_ESA_P		
	PMP_ESB_P		
	PMP_SES_P		
	PMP_SEFS_P		
	PMP_CSS_P		
	PMP_UAS_P		
	PMP_ES		
LR_VT1_5_and_TU11_VC11			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet			
LR_Ten_Gigabit_Ethernet			
LR_Forty_Gigabit_Ethernet			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFINUCASTPKTS		
	PMP_IFINMULTICASTPKTS		
	PMP_IFINBROADCASTPKTS		
	PMP_IFINDISCARDS		
	PMP_IFOUTDISCARDS		
	PMP_IFOUTMULTICASTPKTS		
	PMP_IFOUTBROADCASTPKTS		
	PMP_ETHERSTATSBROADCASTPKTS		
	PMP_ETHERSTATSMULTICASTPKTS		
	PMP_PORTDROPCOUNTS		
	PMP_IFINERRORS		
	PMP_IFOUTOCTETS		
	PMP_IFOUTUCASTPKTS		
	PMP_TXPACKETS		
	PMP_RXPACKETS		
	PMP_RXPAUSEFRAMES		
	PMP_TXPAUSEFRAMES		
	PMP_RXPKTSDROPPEDINTERNALCONGESTION		
	PMP_TXPKTSDROPPEDINTERNALCONGESTION		
	PMP_HDLCPKTDROPS		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATED		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSTXFRAMESBADCRC		
	PMP_RXCONTROLFRAMES		
	PMP_RXUNKNOWNOPCODEFRAMES		
	PMP_IFINERRORBYTEPKTS		
	PMP_IFINFRAMINGERRORPKTS		
	PMP_IFINJUNKINTERPKTS		
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_MEDIAINDSTATSOVERSIZEDROPPED	PML_NEAR_END_RX	15 min, 24 hr
	PMP_DOT3STATSALIGNMENTERRORS		
	PMP_DOT3STATSFCSERRORS		
	PMP_DOT3STATSSINGLECOLLISIONFRAMES		
	PMP_DOT3STATSCARRIERSENSEERRORS		
	PMP_DOT3STATSSQETESTERRORS		
	PMP_DOT3STATSMULTIPLECOLLISIONFRAMES		
	PMP_DOT3STATSDEFERREDTRANSMISSIONS		
	PMP_DOT3STATSLATECOLLISIONS		
	PMP_DOT3STATSEXCESSIVECOLLISIONS		
	PMP_DOT3STATSFRAMETOOLONG		
	PMP_ETHERSTATSUNDERSIZEPKTS		
	PMP_ETHERSTATSFRAGMENTS		
	PMP_ETHERSTATSPKTS64OCTETS		
	PMP_ETHERSTATSPKTS65TO127OCTETS		
	PMP_ETHERSTATSPKTS128TO255OCTETS		
	PMP_ETHERSTATSPKTS256TO511OCTETS		
	PMP_ETHERSTATSPKTS512TO1023OCTETS		
	PMP_ETHERSTATSPKTS1024TO1518OCTETS		
	PMP_ETHERSTATSOVERSIZEPKTS		
	PMP_ETHERSTATSJABBERS		
	PMP_ETHERSTATSOCTETS		
	PMP_ETHERSTATSCOLLISIONS		
	PMP_ETHERSTATSCOLLISIONFRAMES		
	PMP_ETHERSTATSCRCALIGNERRORS		
	PMP_RXETHERUTILIZATIONSTATS		
	PMP_TXETHERUTILIZATIONSTATS		
	PMP_RXGIANTS		
	PMP_RXGMACDROPCOUNTS		
	PMP_RXRUNTS		
	PMP_RXTHRESHOLDOVERSIZES		
	PMP_RXTOTALERRORS		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_TXCOLLISIONS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_TXGIANTS		
	PMP_DOT3STATSCONTROLINUNKNOWNOPCODES		
	PMP_DOT3STATSINPAUSEFRAMES		
	PMP_DOT3STATSOUTPAUSEFRAMES		
	PMP_ETHERSTATSPKTS1519TO1522OCTETS		
LR_POS_100M			
LR_POS_Gigabit			
PTP	PMP_RX_INPUT_DROP_PACKETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RX_INPUT_ABORT_PACKETS		
	PMP_TX_PRE_HDLC_BYTES		
	PMP_TX_POST_HDLC_BYTES		
	PMP_TX_PACKETS		
	PMP_PORT_DROP_COUNTS		
	PMP_RX_PRE_HDLC_BYTES		
	PMP_RX_POST_HDLC_BYTES		
	PMP_RX_PACKETS		
	PMP_RX_NORMAL_PACKETS		
	PMP_RX_SHORTS		
	PMP_RX_RUNTS		
	PMP_RX_LONGS		
	PMP_RX_TOTAL_ERRORS		
	PMP_RX_CRC_ERRORS		
	PMP_ETHERSTATSDROPEVENTS		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATED		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSRXSHORTPKTS		
	PMP_HDLCINOCTETS		
	PMP_HDLCRXABORTS		
	PMP_HDLCOUTOCTETS		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFOUTOCTETS		
	PMP_IFINDISCARDS		
	PMP_RXPKTSDROPINTERNALCONGESTION		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSCSFRAISED		
	PMP_GFPSTATSLFDRAISED		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		
LR_COS_100M			
LR_COS_Gigabit			
PTP	PMP_INTERFACE_NAME	PML_NEAR_END_RX	15 min, 24 hr
	PMP_INTERFACE_DIRECTION		
	PMP_CLASS_OF_SERVICE_LEVEL		
	PMP_POST_POLICY_PACKETS		
	PMP_POST_POLICY_BYTES		
	PMP_DROP_PACKETS		
	PMP_DROP_BYTES		
LR_T1_and_DS1_1_5M			
Note VT1.5 data is retrieved using this layer rate. STS data retrieval is not supported.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
LR_GFP_Multirate			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_GFPSTATSCSFRAISED	PML_NEAR_END_RX	15 min, 24 hr
	PMP_GFPSTATSLFDRAISED		
	PMP_GFPSTATSROUNDTRIPLATENCY		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXSBLKCRCERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		
	PMP_GFPSTATSRXDISTANCEEXTBUFFERS		
	PMP_GFPSTATSTXDISTANCEEXTBUFFERS		
LR_E1_2M			
Note VT2 data is retrieved using this layer rate.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
LR_E1_2M			
Note E1 data is retrieved using this layer rate.			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_CV_P		
	PMP_BBE_P		
	PMP_ES_P		
	PMP_SES_P		
	PMP_UAS_P		
	PMP_SAS_P		
	PMP_FC_P		
	PMP_AISS_P		
LR_OCH_ISC_PEERCOMPAT			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_RXOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RXPACKETS		
	PMP_TXOCTETS		
	PMP_TXPACKETS		
	PMP_INVLD_ORDRD_SET_DISP_ERROR_SUM		
LR_ESCON			
PTP	PMP_IFINERRORS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_TXPACKETS		
	PMP_IFOUTDISCARDS		
	PMP_IFINDISCARDS		
	PMP_IFOUTOCTETS		
	PMP_RXPACKETS		
	PMP_IFINOCTETS		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_TXETHERUTILIZATIONSTATS		
	MP_RXETHERUTILIZATIONSTATS		
	PMP_RXOCTETS		
	PMP_RXPACKETS		
	PMP_TXOCTETS		
	PMP_TXPACKETS		
LR_LEX			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_UAS		
	PMP_SES		
	PMP_ES		
	PMP_FC		
	PMP_NPJC_PGEN		
	PMP_NPJC_PDET		
	PMP_PJCS_PDET		
	PMP_PJCS_PGEN		
	PMP_PJCDIFF		
LR_RPR_IEEE_802_17			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_SPANOUTMCASTCLASSAOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_SPANOUTCTRLFRAMES		
	PMP_SPANOUTOAMECHOFRAMES		
	PMP_SPANOUTOAMFLUSHFRAMES		
	PMP_SPANOUTOAMORGFRAMES		
	PMP_SPANOUTTOPOATDFRAMES		
	PMP_SPANOUTTOPOCHKSUMFRAMES		
	PMP_SPANOUTTOPOTPFRAMES		
	PMP_CLIENTINUCASTCLASSCFRAMES		
	PMP_CLIENTINUCASTCLASSCOCTETS		
	PMP_CLIENTINMCASTCLASSCFRAMES		
	PMP_CLIENTINMCASTCLASSCOCTETS		
	PMP_CLIENTINUCASTCLASSBEIRFRAMES		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_CLIENTINUCASTCLASSBEIROCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_CLIENTINMCASTCLASSBEIRFRAMES		
	PMP_CLIENTINMCASTCLASSBEIROCTETS		
	PMP_CLIENTINUCASTCLASSBCIRFRAMES		
	PMP_CLIENTINUCASTCLASSBCIROCTETS		
	PMP_CLIENTINMCASTCLASSBCIRFRAMES		
	PMP_CLIENTINMCASTCLASSBCIROCTETS		
	PMP_CLIENTINUCASTCLASSAFRAMES		
	PMP_CLIENTINUCASTCLASSAOCTETS		
	PMP_CLIENTINMCASTCLASSAFRAMES		
	PMP_CLIENTINMCASTCLASSAOCTETS		
	PMP_CLIENTINBCASTFRAMES		
	PMP_CLIENTOUTUCASTCLASSCFRAMES		
	PMP_CLIENTOUTUCASTCLASSCOCTETS		
	PMP_CLIENTOUTMCASTCLASSCFRAMES		
	PMP_CLIENTOUTMCASTCLASSCOCTETS		
	PMP_CLIENTOUTUCASTCLASSBEIRFRAMES		
	PMP_CLIENTOUTUCASTCLASSBEIROCTETS		
	PMP_CLIENTOUTMCASTCLASSBEIRFRAMES		
	PMP_CLIENTOUTMCASTCLASSBEIROCTETS		
	PMP_CLIENTOUTUCASTCLASSBCIRFRAMES		
	PMP_CLIENTOUTUCASTCLASSBCIROCTETS		
	PMP_CLIENTOUTMCASTCLASSBCIRFRAMES		
	PMP_CLIENTOUTMCASTCLASSBCIROCTETS		
	PMP_CLIENTOUTUCASTCLASSAFRAMES		
	PMP_CLIENTOUTUCASTCLASSAOCTETS		
	PMP_CLIENTOUTMCASTCLASSAFRAMES		
	PMP_CLIENTOUTMCASTCLASSAOCTETS		
	PMP_CLIENTOUTBCASTFRAMES		
	PMP_ERRORBADPARITYFRAMES		
	PMP_ERRORBADHECFRAMES		
	PMP_ERRORTTLEXPFRAMES		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_ERRORTOOLONGFRAMES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ERRORTOOSHORTFRAMES		
	PMP_ERRORBADFCSFRAMES		
	PMP_ERRORSELFSRCUCASTFRAMES		
	PMP_ERRORPMDABORTFRAMES		
	PMP_ERRORBADADDRFRAMES		
	PMP_ERRORCONTAINEDFRAMES		
	PMP_ERRORSCFFERRORS		
	PMP_ERROROVERSIZEFRAMES		
	PMP_PORTCOUNTERERROR		
	PMP_SPANINUCASTCLASSCFRAMES		
	PMP_SPANINUCASTCLASSCOCTETS		
	PMP_SPANINMCASTCLASSCFRAMES		
	PMP_SPANINMCASTCLASSCOCTETS		
	PMP_SPANINUCASTCLASSBEIRFRAMES		
	PMP_SPANINUCASTCLASSBEIROCTETS		
	PMP_SPANINMCASTCLASSBEIRFRAMES		
	PMP_SPANINMCASTCLASSBEIROCTETS		
	PMP_SPANINUCASTCLASSBCIRFRAMES		
	PMP_SPANINUCASTCLASSBCIROCTETS		
	PMP_SPANINMCASTCLASSBCIRFRAMES		
	PMP_SPANINMCASTCLASSBCIROCTETS		
	PMP_SPANINUCASTCLASSAFRAMES		
	PMP_SPANINUCASTCLASSAOCTETS		
	PMP_SPANINMCASTCLASSAFRAMES		
	PMP_SPANINMCASTCLASSAOCTETS		
	PMP_SPANINCTRLFRAMES		
	PMP_SPANINOAMECHOFRAMES		
	PMP_SPANINOAMFLUSHFRAMES		
	PMP_SPANINOAMORGFRAMES		
	PMP_SPANINTOPOATDFRAMES		
	PMP_SPANINTOPOCHKSUMFRAMES		
	PMP_SPANINTOPOTPFRAMES		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_SPANOUTUCASTCLASSCFRAMES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_SPANOUTUCASTCLASSCOCTETS		
	PMP_SPANOUTMCASTCLASSCFRAMES		
	PMP_SPANOUTMCASTCLASSCOCTETS		
	PMP_SPANOUTUCASTCLASSBEIRFRAMES		
	PMP_SPANOUTUCASTCLASSBEIROCTETS		
	PMP_SPANOUTMCASTCLASSBEIRFRAMES		
	PMP_SPANOUTMCASTCLASSBEIROCTETS		
	PMP_SPANOUTUCASTCLASSBCIRFRAMES		
	PMP_SPANOUTUCASTCLASSBCIROCTETS		
	PMP_SPANOUTMCASTCLASSBCIRFRAMES		
	PMP_SPANOUTMCASTCLASSBCIROCTETS		
	PMP_SPANOUTUCASTCLASSAFRAMES		
	PMP_SPANOUTUCASTCLASSAOCTETS		
	PMP_SPANOUTMCASTCLASSAFRAMES		
LR_T3_and_DS3_45M (on DS3XM, DS3E, DS3NE ports when supported by the NE)			
PTP	PMP_CV_P	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_AISS_P		
	PMP_UAS_P		
	PMP_CV_CP	PML_NEAR_END_RX, PML_FAR_END_RX	
	PMP_ES_CP		
	PMP_SES_CP		
	PMP_SAS_CP		
	PMP_AISS_CP		
	PMP_UAS_CP		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T3_and_DS3_45M			
LR_STS1_and_AU3_High_Order_VC3			
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
LR_STS192c_and_VC4_64c			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_FC		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN	PML_NEAR_END_RX	
	PMP_NPJC_PGEN		
	PMP_PJ_DIFF		
	PMP_PJP_SEC		
	PMP_PJN_SEC		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
LR_Section_OC3_STS3c_and_RS_STM1			
LR_Section_OC12_STS12c_and_RS_STM4			
LR_Section_OC48_STS48c_and_RS_STM16			
LR_Section_OC192_STS192c_and_RS_STM64			
LR_Section_OC768_STS768c_and_ RS_STM256			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_SEFS		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
LR_Line_OC192_STS192_and_MS_STM64			
LR_Section_OC768_STS768c_and_ RS_STM256			
PTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_FC		
	PMP_UAS		
	PMP_PSC	PML_NEAR_END_RX	
	PMP_PSD		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PSC_W		
	PMP_PSD_W		
	PMP_PSC_S		
	PMP_PSD_S		
	PMP_PSC_R		
	PMP_PSD_R		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_PHYSICAL_OPTICAL			

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_MAX_LASER_BIAS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ALARM_FAIL		
	PMP_ALARM_DEGRADE_HIGH		
	PMP_ALARM_DEGRADE_LOW		
	PMP_OSC_LOW		
	PMP_OSC_HIGH		
	PMP_OSC_ALARM_FAIL_LOW		
	PMP_OSC_ALARM_FAIL_HIGH		
	PMP_OSC_ALARM_DEGRADE_LOW		
	PMP_OSC_ALARM_DEGRADE_HIGH		
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_Optical_Channel			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
LR_PHYSICAL_OPTICAL			
PTP	PMP_MIN_TX_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_TX_POWER		
	PMP_MAX_TX_POWER		
	PMP_MIN_RX_POWER		
	PMP_AVG_RX_POWER		
	PMP_MAX_RX_POWER		
	PMP_MIN_LASER_BIAS		
	PMP_AVG_LASER_BIAS		
	PMP_MAX_LASER_BIAS		
	PMP_MIN_TX_OSC_POWER		
	PMP_AVG_TX_OSC_POWER		
	PMP_MAX_TX_OSC_POWER		
LR_Optical_Channel			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
PTP	PMP_MIN_PT_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_PT_POWER		
	PMP_MAX_PT_POWER		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_T3_and_DS3_45M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_AISSP_PBIT	PML_NEAR_END_RX	
	PMP_CVP_PBIT		
	PMP_ESP_PBIT		
	PMP_SASP_PBIT		
	PMP_SESP_PBIT		
	PMP_UASP_PBIT		
	PMP_CVP_CPBIT	PML_NEAR_END_RX, PML_FAR_END_RX	
	PMP_ESP_CPBIT		
	PMP_SESP_CPBIT		
	PMP_UASP_CPBIT		
	PMP_SASP_CPBIT	PML_FAR_END_RX	
LR_T1_and_DS1_1_5M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
PTP	PMP_CV_P	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_UAS_P		
	PMP_AISS_P		
	PMP_CV_P	PML_FAR_END_RX	
	PMP_ES_P		
	PMP_ESA_P		
	PMP_ESB_P		
	PMP_SES_P		
	PMP_SEFS_P		
	PMP_CSS_P		
	PMP_UAS_P		
	PMP_ES		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Optical_Transport_Section			
PTP	PMP_ES	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_SES		
	PMP_UAS		
	PMP_BBE		
	PMP_FC		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_Optical_Transport_Path			
PTP	PMP_ES_P	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_SES_P		
	PMP_UAS_P		
	PMP_BBE_P		
	PMP_FC_P		
	PMP_ESR_P		
	PMP_SESR_P		
	PMP_BBER_P		
LR_Optical_Transport_FEC			
PTP	PMP_BIT_ERRORS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_UNCORRECTABLEWORDS		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_FC_100_1063_M			
LR_FC_200_2125M			
LR_FICON_1G			
LR_FICON_2G			
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RXTOTALPKTS		
	PMP_IFINDISCARDS		
	PMP_IFINERRORS		
	PMP_IFOUTOCTETS		
	PMP_TXTOTALPKTS		
	PMP_IFOUTDISCARDS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSRXSBLKCRCERRORS		
	PMP_GFPSTATSCSFRAISED		
	PMP_GFPSTATSROUNDTRIPLATENCY		
	PMP_FCINGRESSRXDISTANCEEXTBUFFERS		
	PMP_FCEGRESSTXDISTANCEEXTBUFFERS		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSTXFRAMESBADCRC		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATED		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
LR_FC_100_1063_M			
LR_FC_200_2125M			
LR_FICON_1G			
LR_FICON_2G (continued)			
PTP	PMP_FCSTATSLINKRECOVERIES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_FCSTATSRXCREDITS		
	PMP_FCSTATSTXCREDITS		
	PMP_FCSTATSZEROTXCREDITS		
	PMP_8B10BINVALIDORDEREDSETS		
	PMP_8B10BSTATSENCODINGDISPERRORS		
	PMP_RXUTILIZATION		
	PMP_TXUTILIZATION		

Table 3-7 ONS 15454 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_8b10b_2_5_Gigabit			
PTP	PMP_DATA_PAYLOAD	PML_NEAR_END_RX	15 min, 24 hr
	PMP_VALID_PACKETS		
	PMP_INVALID_PACKETS		
	PMP_CODE_GROUP_VIOLATIONS		
	PMP_IDLE_ORDERED_SETS		
	PMP_NON_IDLE_ORDERED_SETS		
	PMP_DATA_CODE_GROUPS		
	PMP_RXTOTALPACKETS		
	PMP_IFINERRORS		
	PMP_STATS_ENCODING_DISPERRORS		
	PMP_DATA_ORDERED_SETS		
LR_VT1_5_and_TU11_VC11			
CTP	Same as ONS 15327	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
LR_E1_2M (only on DS1_E1_56)			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_CV_P		
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_AISS_P		
	PMP_UAS_P		
	PMP_FC_P		
	PMP_BBE_P		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet			
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFINUCASTPKTS		
	PMP_IFINMULTICASTPKTS		
	PMP_IFINBROADCASTPKTS		
	PMP_IFINDISCARDS		
	PMP_IFOUTDISCARDS		
	PMP_IFOUTMULTICASTPKTS		
	PMP_IFOUTBROADCASTPKTS		
	PMP_ETHERSTATSBROADCASTPKTS		
	PMP_ETHERSTATSMULTICASTPKTS		
	PMP_PORTDROPCOUNTS		
	PMP_IFINERRORS		
	PMP_IFOUTOCTETS		
	PMP_IFOUTUCASTPKTS		
	PMP_TXPACKETS		
	PMP_RXPACKETS		
	PMP_RXPAUSEFRAMES		
	PMP_TXPAUSEFRAMES		
	PMP_RXPKTSDROPPEDINTERNALCONGESTION		
	PMP_TXPKTSDROPPEDINTERNALCONGESTION		
	PMP_HDLCPKTDROPS		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATED		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSTXFRAMESBADCRC		
	PMP_RXCONTROLFRAMES		
	PMP_RXUNKNOWNOPCODEFRAMES		
	PMP_IFINERRORBYTEPKTS		
	PMP_IFINFRAMINGERRORPKTS		
	PMP_IFINJUNKINTERPKTS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			
PTP	PMP_MEDIAINDSTATSOVERSIZEDROPPED	PML_NEAR_END_RX	15 min, 24 hr
	PMP_DOT3STATSALIGNMENTERRORS		
	PMP_DOT3STATSFCSERRORS		
	PMP_DOT3STATSSINGLECOLLISIONFRAMES		
	PMP_DOT3STATSCARRIERSENSEERRORS		
	PMP_DOT3STATSSQETESTERRORS		
	PMP_DOT3STATSMULTIPLECOLLISIONFRAMES		
	PMP_DOT3STATSDEFERREDTRANSMISSIONS		
	PMP_DOT3STATSLATECOLLISIONS		
	PMP_DOT3STATSEXCESSIVECOLLISIONS		
	PMP_DOT3STATSFRAMETOOLONG		
	PMP_ETHERSTATSUNDERSIZEPKTS		
	PMP_ETHERSTATSFRAGMENTS		
	PMP_ETHERSTATSPKTS64OCTETS		
	PMP_ETHERSTATSPKTS65TO127OCTETS		
	PMP_ETHERSTATSPKTS128TO255OCTETS		
	PMP_ETHERSTATSPKTS256TO511OCTETS		
	PMP_ETHERSTATSPKTS512TO1023OCTETS		
	PMP_ETHERSTATSPKTS1024TO1518OCTETS		
	PMP_ETHERSTATSOVERSIZEPKTS		
	PMP_ETHERSTATSJABBERS		
	PMP_ETHERSTATSOCTETS		
	PMP_ETHERSTATSCOLLISIONS		
	PMP_ETHERSTATSCOLLISIONFRAMES		
	PMP_ETHERSTATSCRCALIGNERRORS		
	PMP_RXETHERUTILIZATIONSTATS		
	PMP_TXETHERUTILIZATIONSTATS		
	PMP_RXGIANTS		
	PMP_RXGMACDROPCOUNTS		
	PMP_RXRUNTS		
	PMP_RXTHRESHOLDOVERSIZES		
	PMP_RXTOTALERRORS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Fast_Ethernet			
LR_Gigabit_Ethernet (continued)			
PTP	PMP_TXCOLLISIONS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_TXGIANTS		
	PMP_DOT3STATSCONTROLINUNKNOWNOPCODES		
	PMP_DOT3STATSINPAUSEFRAMES		
	PMP_DOT3STATSOUTPAUSEFRAMES		
	PMP_ETHERSTATSPKTS1519TO1522OCTETS		
LR_POS_100M			
LR_POS_Gigabit			
PTP	PMP_RX_INPUT_DROP_PACKETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RX_INPUT_ABORT_PACKETS		
	PMP_TX_PRE_HDLC_BYTES		
	PMP_TX_POST_HDLC_BYTES		
	PMP_TX_PACKETS		
	PMP_PORT_DROP_COUNTS		
	PMP_RX_PRE_HDLC_BYTES		
	PMP_RX_POST_HDLC_BYTES		
	PMP_RX_PACKETS		
	PMP_RX_NORMAL_PACKETS		
	PMP_RX_SHORTS		
	PMP_RX_RUNTS		
	PMP_RX_LONGS		
	PMP_RX_TOTAL_ERRORS		
	PMP_RX_CRC_ERRORS		
	PMP_ETHERSTATSDROPEVENTS		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATED		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_MEDIAINDSTATSRXSHORTPKTS		
	PMP_HDLCINOCTETS		
	PMP_HDLCRXABORTS		
	PMP_HDLCOUTOCTETS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_POS_100M			
LR_POS_Gigabit (continued)			
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFOUTOCTETS		
	PMP_IFINDISCARDS		
	PMP_RXPKTSDROPINTERNALCONGESTION		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSCSFRAISED		
	PMP_GFPSTATSLFDRAISED		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		
LR_COS_100M			
LR_COS_Gigabit			
PTP	PMP_INTERFACE_NAME	PML_NEAR_END_RX	15 min, 24 hr
	PMP_INTERFACE_DIRECTION		
	PMP_CLASS_OF_SERVICE_LEVEL		
	PMP_POST_POLICY_PACKETS		
	PMP_POST_POLICY_BYTES		
	PMP_DROP_PACKETS		
	PMP_DROP_BYTES		
LR_T1_and_DS1_1_5M			
Note VT1.5 data is retrieved using this layer rate. STS data retrieval is not supported.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_GFP_Multirate			
PTP	PMP_GFPSTATSCSFRAISED	PML_NEAR_END_RX	15 min, 24 hr
	PMP_GFPSTATSLFDRAISED		
	PMP_GFPSTATSROUNDTRIPLATENCY		
	PMP_GFPSTATSRXCIDINVALID		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXOCTETS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXSBLKCRCERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_GFPSTATSTXFRAME		
	PMP_GFPSTATSTXOCTETS		
	PMP_GFPSTATSRXDISTANCEEXTBUFFERS		
	PMP_GFPSTATSTXDISTANCEEXTBUFFERS		
LR_E1_2M			
Note VT2 data is retrieved using this layer rate.			
CTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
LR_E1_2M			
Note E1 data is retrieved using this layer rate.			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_CV_P		
	PMP_BBE_P		
	PMP_ES_P		
	PMP_SES_P		
	PMP_UAS_P		
	PMP_SAS_P		
	PMP_FC_P		
	PMP_AISS_P		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_OCH_ISC_PEERCOMPAT			
PTP	PMP_RXOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_RXPACKETS		
	PMP_TXOCTETS		
	PMP_TXPACKETS		
	PMP_INVLD_ORDRD_SET_DISP_ERROR_SUM		
LR_ESCON			
PTP	PMP_IFINERRORS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_TXPACKETS		
	PMP_IFOUTDISCARDS		
	PMP_IFINDISCARDS		
	PMP_IFOUTOCTETS		
	PMP_RXPACKETS		
	PMP_IFINOCTETS		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_TXETHERUTILIZATIONSTATS		
	MP_RXETHERUTILIZATIONSTATS		
	PMP_RXOCTETS		
	PMP_RXPACKETS		
	PMP_TXOCTETS		
	PMP_TXPACKETS		
LR_LEX			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_UAS		
	PMP_SES		
	PMP_ES		
	PMP_FC		
	PMP_NPJC_PGEN		
	PMP_NPJC_PDET		
	PMP_PJCS_PDET		
	PMP_PJCS_PGEN		
	PMP_PJCDIFF		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_RPR_IEEE_802_17			
PTP	PMP_SPANOUTMCASTCLASSAOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_SPANOUTCTRLFRAMES		
	PMP_SPANOUTOAMECHOFRAMES		
	PMP_SPANOUTOAMFLUSHFRAMES		
	PMP_SPANOUTOAMORGFRAMES		
	PMP_SPANOUTTOPOATDFRAMES		
	PMP_SPANOUTTOPOCHKSUMFRAMES		
	PMP_SPANOUTTOPOTPFRAMES		
	PMP_CLIENTINUCASTCLASSCFRAMES		
	PMP_CLIENTINUCASTCLASSCOCTETS		
	PMP_CLIENTINMCASTCLASSCFRAMES		
	PMP_CLIENTINMCASTCLASSCOCTETS		
	PMP_CLIENTINUCASTCLASSBEIRFRAMES		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_RPR_IEEE_802_17 (continued)			
PTP	PMP_CLIENTINUCASTCLASSBEIROCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_CLIENTINMCASTCLASSBEIRFRAMES		
	PMP_CLIENTINMCASTCLASSBEIROCTETS		
	PMP_CLIENTINUCASTCLASSBCIRFRAMES		
	PMP_CLIENTINUCASTCLASSBCIROCTETS		
	PMP_CLIENTINMCASTCLASSBCIRFRAMES		
	PMP_CLIENTINMCASTCLASSBCIROCTETS		
	PMP_CLIENTINUCASTCLASSAFRAMES		
	PMP_CLIENTINUCASTCLASSAOCTETS		
	PMP_CLIENTINMCASTCLASSAFRAMES		
	PMP_CLIENTINMCASTCLASSAOCTETS		
	PMP_CLIENTINBCASTFRAMES		
	PMP_CLIENTOUTUCASTCLASSCFRAMES		
	PMP_CLIENTOUTUCASTCLASSCOCTETS		
	PMP_CLIENTOUTMCASTCLASSCFRAMES		
	PMP_CLIENTOUTMCASTCLASSCOCTETS		
	PMP_CLIENTOUTUCASTCLASSBEIRFRAMES		
	PMP_CLIENTOUTUCASTCLASSBEIROCTETS		
	PMP_CLIENTOUTMCASTCLASSBEIRFRAMES		
	PMP_CLIENTOUTMCASTCLASSBEIROCTETS		
	PMP_CLIENTOUTUCASTCLASSBCIRFRAMES		
	PMP_CLIENTOUTUCASTCLASSBCIROCTETS		
	PMP_CLIENTOUTMCASTCLASSBCIRFRAMES		
	PMP_CLIENTOUTMCASTCLASSBCIROCTETS		
	PMP_CLIENTOUTUCASTCLASSAFRAMES		
	PMP_CLIENTOUTUCASTCLASSAOCTETS		
	PMP_CLIENTOUTMCASTCLASSAFRAMES		
	PMP_CLIENTOUTMCASTCLASSAOCTETS		
	PMP_CLIENTOUTBCASTFRAMES		
	PMP_ERRORBADPARITYFRAMES		
	PMP_ERRORBADHECFRAMES		
	PMP_ERRORTTLEXPFRAMES		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_RPR_IEEE_802_17 (continued)			
PTP	PMP_ERRORTOOLONGFRAMES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ERRORTOOSHORTFRAMES		
	PMP_ERRORBADFCSFRAMES		
	PMP_ERRORSELFSRCUCASTFRAMES		
	PMP_ERRORPMDABORTFRAMES		
	PMP_ERRORBADADDRFRAMES		
	PMP_ERRORCONTAINEDFRAMES		
	PMP_ERRORSCFFERRORS		
	PMP_ERROROVERSIZEFRAMES		
	PMP_PORTCOUNTERERROR		
	PMP_SPANINUCASTCLASSCFRAMES		
	PMP_SPANINUCASTCLASSCOCTETS		
	PMP_SPANINMCASTCLASSCFRAMES		
	PMP_SPANINMCASTCLASSCOCTETS		
	PMP_SPANINUCASTCLASSBEIRFRAMES		
	PMP_SPANINUCASTCLASSBEIROCTETS		
	PMP_SPANINMCASTCLASSBEIRFRAMES		
	PMP_SPANINMCASTCLASSBEIROCTETS		
	PMP_SPANINUCASTCLASSBCIRFRAMES		
	PMP_SPANINUCASTCLASSBCIROCTETS		
	PMP_SPANINMCASTCLASSBCIRFRAMES		
	PMP_SPANINMCASTCLASSBCIROCTETS		
	PMP_SPANINUCASTCLASSAFRAMES		
	PMP_SPANINUCASTCLASSAOCTETS		
	PMP_SPANINMCASTCLASSAFRAMES		
	PMP_SPANINMCASTCLASSAOCTETS		
	PMP_SPANINCTRLFRAMES		
	PMP_SPANINOAMECHOFRAMES		
	PMP_SPANINOAMFLUSHFRAMES		
	PMP_SPANINOAMORGFRAMES		
	PMP_SPANINTOPOATDFRAMES		
	PMP_SPANINTOPOCHKSUMFRAMES		
	PMP_SPANINTOPOTPFRAMES		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_RPR_IEEE_802_17 (continued)			
PTP	PMP_SPANOUTUCASTCLASSCFRAMES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_SPANOUTUCASTCLASSCOCTETS		
	PMP_SPANOUTMCASTCLASSCFRAMES		
	PMP_SPANOUTMCASTCLASSCOCTETS		
	PMP_SPANOUTUCASTCLASSBEIRFRAMES		
	PMP_SPANOUTUCASTCLASSBEIROCTETS		
	PMP_SPANOUTMCASTCLASSBEIRFRAMES		
	PMP_SPANOUTMCASTCLASSBEIROCTETS		
	PMP_SPANOUTUCASTCLASSBCIRFRAMES		
	PMP_SPANOUTUCASTCLASSBCIROCTETS		
	PMP_SPANOUTMCASTCLASSBCIRFRAMES		
	PMP_SPANOUTMCASTCLASSBCIROCTETS		
	PMP_SPANOUTUCASTCLASSAFRAMES		
	PMP_SPANOUTUCASTCLASSAOCTETS		
	PMP_SPANOUTMCASTCLASSAFRAMES		
LR_T3_and_DS3_45M			
PTP	PMP_CV_P	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_AISS_P		
	PMP_UAS_P		
	PMP_CV_CP		
	PMP_ES_CP		
	PMP_SES_CP		
	PMP_SAS_CP		
	PMP_UAS_CP		
	PMP_CV		
	PMP_ES		
	PMP_SES		
	PMP_LOSS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_E1_2M			
LR_E3_34M			
LR_T3_and_DS3_45M			
CTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
LR_STS192c_and_VC4_64c			
CTP	PMP_EB	PML_NEAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
LR_STS192c_and_VC4_64c (continued)			
CTP	PMP_PJ_DIFF	PML_NEAR_END_RX	15 min, 24 hr
	PMP_PJP_SEC		
	PMP_PJN_SEC		
LR_Section_OC3_STS3c_and_RS_STM1			
LR_Section_OC12_STS12c_and_RS_STM4			
LR_Section_OC48_STS48c_and_RS_STM16			
LR_Section_OC192_STS192c_and_RS_STM64			
LR_Section_OC768_STS768c_and_ RS_STM256			
(STM1E card does not support PMP_OFS)			
PTP	PMP_EB	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_BBE		
	PMP_UAS		
	PMP_OFS (STM1E cards do not support PMP_OFS.)		
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
LR_Line_OC192_STS192_and_MS_STM64			
LR_Line_OC768_STS768c_and_ RS_STM256			
PTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_BBE		
	PMP_UAS		
	PMP_FC		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
LR_Line_OC192_STS192_and_MS_STM64			
LR_Line_OC768_STS768c_and_ RS_STM256 (continued)			
PTP	PMP_PSC	PML_NEAR_END_RX	15 min, 24 hr
	PMP_PSD		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PSC_W		
	PMP_PSD_W		
	PMP_PSC_S		
	PMP_PSD_S		
	PMP_PSC_R		
	PMP_PSD_R		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
	PMP_NPJC_PGEN		
	PMP_PPJC_PGEN		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_PHYSICAL_OPTICAL			
PTP	PMP_MAX_LASER_BIAS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ALARM_FAIL		
	PMP_ALARM_DEGRADE_HIGH		
	PMP_ALARM_DEGRADE_LOW		
	PMP_OSC_LOW		
	PMP_OSC_HIGH		
	PMP_OSC_ALARM_FAIL_LOW		
	PMP_OSC_ALARM_FAIL_HIGH		
	PMP_OSC_ALARM_DEGRADE_LOW		
	PMP_OSC_ALARM_DEGRADE_HIGH		
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_Optical_Channel			
LR_PHYSICAL_OPTICAL			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
PTP	PMP_MIN_TX_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_TX_POWER		
	PMP_MAX_TX_POWER		
	PMP_MIN_RX_POWER		
	PMP_AVG_RX_POWER		
	PMP_MAX_RX_POWER		
	PMP_MIN_LASER_BIAS		
	PMP_AVG_LASER_BIAS		
	PMP_MAX_LASER_BIAS		
	PMP_MIN_TX_OSC_POWER		
	PMP_AVG_TX_OSC_POWER		
	PMP_MAX_TX_OSC_POWER		
LR_Optical_Channel			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
PTP	PMP_MIN_PT_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_PT_POWER		
	PMP_MAX_PT_POWER		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_LOW_ORDER_TU_VC3			
LR_VT2_AND_TU12_VC12			
CTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_E4_140M			
PTP	PMP_CV	PML_NEAR_END_RX,	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
	PMP_BBE		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_E3_34M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		

Table 3-8 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_E1_2M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_EB_P	PML_NEAR_END_RX, PML_FAR_END_RX, PML_NEAR_END_TX, PML_FAR_END_TX	
	PMP_BBE_P		
	PMP_ES_P		
	PMP_SES_P		
	PMP_UAS_P		
	PMP_ESR_P		
	PMP_SESR_P		
	PMP_BBER_P		
	PMP_AISS	PML_NEAR_END_RX, PML_NEAR_END_TX	
OTN SECTION			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
OTN PATH			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
OTN FEC			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
SAN			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
8B10B			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_T3_and_DS3_45M			

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
PTP	PMP_CV_P	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES_P		
	PMP_SES_P		
	PMP_SAS_P		
	PMP_AISS_P		
	PMP_UAS_P		
	PMP_CV_CP		
	PMP_ES_CP		
	PMP_SES_CP		
	PMP_SAS_CP		
	PMP_UAS_CP		
	PMP_CV		
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
LR_E1_2M			
LR_E3_34M			
LR_T3_and_DS3_45M			
CTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS6c_and_VC4_2c			
LR_STS9c_and_VC4_3c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
LR_STS192c_and_VC4_64c			

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
CTP	PMP_EB	PML_NEAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
LR_STS3c_and_AU4_VC4 **LR_STS12c_and_VC4_4c** **LR_STS6c_and_VC4_2c** **LR_STS9c_and_VC4_3c** **LR_STS24c_and_VC4_8c** **LR_STS48c_and_VC4_16c** **LR_STS192c_and_VC4_64c (continued)**			
CTP	PMP_PJ_DIFF	PML_NEAR_END_RX	15 min, 24 hr
	PMP_PJP_SEC		
	PMP_PJN_SEC		
LR_Section_OC3_STS3c_and_RS_STM1 **LR_Section_OC12_STS12c_and_RS_STM4** **LR_Section_OC48_STS48c_and_RS_STM16** **LR_Section_OC192_STS192c_and_RS_STM64** **LR_Section_OC768_STS768c_and_ RS_STM256** **(STM1E card does not support PMP_OFS)**			
PTP	PMP_EB	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_BBE		
	PMP_UAS		
	PMP_OFS (STM1E cards do not support PMP_OFS.)		

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
LR_Line_OC192_STS192_and_MS_STM64			
LR_Line_OC768_STS768c_and_ RS_STM256			
PTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_BBE		
	PMP_UAS		
	PMP_FC		
LR_Line_OC3_STS3_and_MS_STM1			
LR_Line_OC12_STS12_and_MS_STM4			
LR_Line_OC48_STS48_and_MS_STM16			
LR_Line_OC192_STS192_and_MS_STM64			
LR_Line_OC768_STS768c_and_ RS_STM256 (continued)			
PTP	PMP_PSC	PML_NEAR_END_RX	15 min, 24 hr
	PMP_PSD		
	PMP_PPJ		
	PMP_NPJ		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		
	PMP_PSC_W		
	PMP_PSD_W		
	PMP_PSC_S		
	PMP_PSD_S		
	PMP_PSC_R		
	PMP_PSD_R		
	PMP_PPJC_PDET		
	PMP_NPJC_PDET		
	PMP_NPJC_PGEN		
	PMP_PPJC_PGEN		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_PHYSICAL_OPTICAL			
PTP	PMP_MAX_LASER_BIAS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ALARM_FAIL		
	PMP_ALARM_DEGRADE_HIGH		
	PMP_ALARM_DEGRADE_LOW		
	PMP_OSC_LOW		
	PMP_OSC_HIGH		
	PMP_OSC_ALARM_FAIL_LOW		
	PMP_OSC_ALARM_FAIL_HIGH		
	PMP_OSC_ALARM_DEGRADE_LOW		
	PMP_OSC_ALARM_DEGRADE_HIGH		
LR_Physical_10_Gigabit_ITU			
LR_Physical_2_5_Gigabit_ITU			
LR_Optical_Channel			
LR_PHYSICAL_OPTICAL			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
PTP	PMP_MIN_TX_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_TX_POWER		
	PMP_MAX_TX_POWER		
	PMP_MIN_RX_POWER		
	PMP_AVG_RX_POWER		
	PMP_MAX_RX_POWER		
	PMP_MIN_LASER_BIAS		
	PMP_AVG_LASER_BIAS		
	PMP_MAX_LASER_BIAS		
LR_Optical_Channel			
LR_Optical_Multiplex_Section			
LR_Optical_Transmission_Section			
PTP	PMP_MIN_PT_POWER	PML_NEAR_END_RX	15 min, 24 hr
	PMP_AVG_PT_POWER		
	PMP_MAX_PT_POWER		
LR_LOW_ORDER_TU_VC3			
LR_VT2_AND_TU12_VC12			

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
CTP	PMP_EB	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_BBE		
	PMP_SES		
	PMP_ES		
	PMP_UAS		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_E4_140M			
PTP	PMP_CV	PML_NEAR_END_RX,	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
	PMP_BBE		
	PMP_ESR		
	PMP_SESR		
	PMP_BBER		
LR_E3_34M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
LR_E1_2M			
PTP	PMP_CV	PML_NEAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_LOSS		
	PMP_EB_P	PML_NEAR_END_RX, PML_FAR_END_RX, PML_NEAR_END_TX, PML_FAR_END_TX	
	PMP_BBE_P		
	PMP_ES_P		
	PMP_SES_P		
	PMP_UAS_P		
	PMP_ESR_P		
	PMP_SESR_P		
	PMP_BBER_P		
	PMP_AISS	PML_NEAR_END_RX, PML_NEAR_END_TX	

Table 3-9 ONS 15454 SDH TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
OTN SECTION			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
OTN PATH			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
OTN FEC			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
SAN			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr
8B10B			
PTP	Same as SONET	PML_NEAR_END_RX	15 min, 24 hr

Table 3-10 ONS 15600 SONET TCA TP ParameterName Values

PTP or CTP	pmParameterName	Location	Interval
LR_Gigabit_Ethernet			
PTP	PMP_IFINOCTETS	PML_NEAR_END_RX	15 min, 24 hr
	PMP_IFINUCASTPKTS		
	PMP_IFINMULTICASTPKTS		
	PMP_IFINBROADCASTPKTS		
	PMP_ETHERSTATSOVERSIZEPKTS		
	PMP_DOT3STATSFCSERRORS		
	PMP_ETHERSTATSUNDERSIZEPKTS		
	PMP_ETHERSTATSJABBERS		
	PMP_DOT3STATSALIGNMENTERRORS		
	PMP_IFOUTOCTETS		
	PMP_IFOUTUCASTPKTS		
	PMP_IFOUTMULTICASTPKTS		
	PMP_IFOUTBROADCASTPKTS		
	PMP_ETHERSTATSDROPEVENTS		
	PMP_IFINDISCARDS		
	PMP_ETHERSTATSOCTETS		
	PMP_ETHERSTATSPKTS		
	PMP_ETHERSTATSBROADCASTPKTS		
	PMP_ETHERSTATSMULTICASTPKTS		
	PMP_ETHERSTATSMULTICASTPKTS		
	PMP_ETHERSTATSFRAGMENTS		
	PMP_ETHERSTATSPKTS64OCTETS		

Table 3-10 ONS 15600 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
	PMP_ETHERSTATSPKTS65TO127OCTETS		
	PMP_ETHERSTATSPKTS128TO255OCTETS		
	PMP_ETHERSTATSPKTS256TO511OCTETS		
	PMP_ETHERSTATSPKTS512TO1023OCTETS		
	PMP_ETHERSTATSPKTS1024TO1518OCTETS		
	PMP_IFOUTDISCARDS		
	PMP_IFINERRORS		
	PMP_IFOUTERRORS		
	PMP_DOT3STATSINTERNALMACTXERRORS		
	PMP_DOT3STATSFRAMETOOLONG		
	PMP_DOT3STATSINTERNALMACRXERRORS		
	PMP_DOT3STATSSYMBOLERRORS		
	PMP_RX_PAUSE_FRAMES		
	PMP_TX_PAUSE_FRAMES		
	PMP_RXUTILIZATION		
	PMP_TXUTILIZATION		
LR_POS_Gigabit			
PTP	PMP_GFPSTATSRXOCTETS	PML_NEAR_END_RX	15 min, 24 hr
PTP	PMP_RX_PACKETS		
	PMP_GFPSTATSRXCRCERRORS		
	PMP_GFPSTATSTXOCTETS		
	PMP_TX_PACKETS		
	PMP_GFPSTATSRXSBITERRORS		
	PMP_GFPSTATSRXMBITERRORS		
	PMP_GFPSTATSRXTYPEINVALID		
	PMP_HDLCRXABORTS		
	PMP_MEDIAINDSTATSRXSHORTPKTS		
	PMP_MEDIAINDSTATSRXFRAMESTOOLONG		
	PMP_RXPKTSDROPPEDINERNALCONGESTION		
	PMP_MEDIAINDSTATSRXFRAMESTRUNCATE D		
	PMP_MEDIAINDSTATSRXFRAMESBADCRC		
	PMP_GFPSTATSRXFRAME		
	PMP_GFPSTATSTXFRAME		
	PMP_HDLCINOCTETS		
	PMP_HDLCOUTOCTETS		

Table 3-10 ONS 15600 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
	PMP_RXUTILIZATION		
LR_PHYSICAL_OPTICAL			
PTP	PMP_LBC_HIGH PMP_LBC_LOW PMP_OPT_HIGH PMP_OPR_HIGH PMP_OPR_LOW	PML_NEAR_END_RX	15 min, 24 hr
LR_Section_OC48_STS48c_and_RS_STM16 **LR_Section_OC192_STS192_and_RS_STM64**			
PTP	PMP_ES	PML_NEAR_END_RX	15 min, 24 hr
	PMP_SES		
	PMP_SEFS		
	PM_CV		
LR_Line_OC48_STS48_and_MS_STM16 **LR_Line_OC192_STS192_and_MS_STM64**			
PTP	PMP_CV	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hr
	PMP_ES		
	PMP_SES		
	PMP_UAS		
	PMP_FC		
	PMP_PSC	PML_NEAR_END_RX	
	PMP_PSD		
	PMP_PSC_W		
	PMP_PSD_W		
	PMP_PSC_S		
	PMP_PSD_S		
	PMP_PSC_R		
	PMP_PSD_R		

Table 3-10 ONS 15600 SONET TCA TP ParameterName Values (continued)

PTP or CTP	pmParameterName	Location	Interval
LR_STS1_and_AU3_High_Order_VC3			
LR_STS3c_and_AU4_VC4			
LR_STS12c_and_VC4_4c			
LR_STS24c_and_VC4_8c			
LR_STS48c_and_VC4_16c			
LR_STS192c_and_VC4_64c			
CTP	PMP_CV_P	PML_NEAR_END_RX, PML_FAR_END_RX	15 min, 24 hour
	PMP_ES_P		
	PMP_SES_P		
	PMP_UAS_P		
	PMP_FC_P		
	PMP_PPJC_PDET	PML_NEAR_END_RX	15 min, 24 hour
	PMP_NPJC_PDET		
	PMP_PPJC_PGEN		
	PMP_NPJC_PGEN		

- PMLocation_T pmLocation

  CTM supports PML_NEAR_END_RX and PML_FAR_END_RX.
- PMThresholdType_T thresholdType

  CTM supports only TWM_HIGH.
- boolean triggerFlag

  Not supported.
- float value

  Supported. The NE accepts only integer values. Therefore, CTM rounds off floating point values to the biggest lower integer. The maximum value supported is 9999999.
- string unit

  Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if tpName does not reference a terminationPoint object or at
least one field of tcaParameters is not valid.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.13.4 getTCATPParameter

Synopsis

```
void getTCATPParameter(
in globaldefs::NamingAttributes_T tpName,
in transmissionParameters::LayerRate_T layerRate,
in Granularity_T granularity,
out TCAParameters_T tcaParameter)
raises(globaldefs::ProcessingFailureException);
```

Description

The purpose of this operation is to retrieve the values of 15-minute or 24-hour PM thresholds on a TP/layer rate measurement point. The operation is best-effort. Results are returned in the out parameter of this operation. The operation can be applied to either a PTP or CTP. The NMS requests the TCA parameters for the particular TP and granularity specified.

Parameters

Name	Type	Input/Output	Description
tpName	globaldefs::NamingAttributes_T	In	Identification of the TP for which to retrieve the values of the TCA parameters.
layerRate	transmissionParameters::LayerRate_T	In	Layer rate for which to retrieve the values of the TCA parameters.
granularity	Granularity_T	In	Granularity for which to retrieve the TCA parameters.
tcaParameter	TCAParameters_T	Out	Result of the operation.

For a list of supported layer rates and parameter values, see 3.11.3 getAllVLANs, page 3-207.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tpName does not reference a terminationPoint object or
layerRateList contains undefined rates or granularity contains an undefined value.

EXCPT_ENTITY_NOT_FOUND - Raised when tpName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.13.5 getCTMHistoryPMData

Synopsis

```
    public void getCTMHistoryPMData(
        in nmsSession::NmsSession_I client,
        in PMTPSelectList_T pmTPSelectList,
        in PMParameterNameList_T pmParameters,
        in globaldefs::Time_T startTime,
        in globaldefs::Time_T endTime)
raises (globaldefs::ProcessingFailureException);
```

Description

This interface instructs the EMS to store historical PM data in a file, and to notify the NMS when the request is complete. Within the request a list of TP/layerRate measurement points and a time frame are specified. For each measurement point, the granularity (15-minute, 24-hour) and location (PML_NEAR_END_RX, PML_FAR_END_RX) can be specified. A filtered set (scoped by the input parameter pmParameters) of PM parameters collected for a particular TP/layer tate measurement point for the specified granularity, location, and time window is made available.

Measurement intervals and the time frame are considered as half-open intervals to the right (that is, startTime <= t < endTime).

Performance monitoring data on multiple TPs of multiple MEs is transferred in one data file. The PM file format is defined by the TMF. CTM generates a unique filename and saves the file in the *CTMS-home/*pmData*/username/* directory. This file is retained for six hours after CTM finishes writing the requested PM data. After six hours, the file is deleted. When CTM GateWay/CORBA restarts, it retrieves a list of all existing PM files and deletes them after six hours.

This is an asynchronous operation. Once CTM receives the request, it validates it and the call returns. In the background, CTM begins writing to the file. Upon successful completion, it notifies the NMS session by invoking nmsSession::NmsSession_I::historyPMDataCompleted. If CTM fails to create this file or write into the file, the NMS is notified by invoking nmsSession::NmsSession_I::historyPMDataFailed.

CTM allows only one request from each NMS session at a time.

For ONS 15454 SDH NEs, this operation is not supported on low-order STM-path, but is supported on path parameters on E3 and DS3 cards.

For the DS1 module, this operation is not supported on STS/HO path, but is supported on VT data (SONET) and VC11 data (SDH).

For the E1 module, this operation is not supported on STS/HO path, but is supported on VT data (SONET) and VC12 data (SDH).

For the E3 module, this operation is not supported on HO path, but is supported on VC3 data (SDH).

For the DS3 module, this operation is not supported on HO path, but is supported on VC3 data (SDH). CTM raises an EXCPT_UNABLE_TO_COMPLY exception if another request comes from a session for which one request is already in progress. At any given time, CTM has a maximum of eight such requests. This restriction is imposed because multiple requests downgrade the CTM database performance and have an adverse impact on CTM.

Relevance of the validity status for historical PM data and the one-to-one mapping from PM data on the EMS (CTM) to CTM GateWay/CORBA:

- –1—The PM parameter is not applicable to the module in that particular physical location (N/A). The validity status is Invalid.
- –2—The PM data bucket is empty for the parameters in the module, for a time interval (blank). The validity status is Invalid.

- –3—The module in the physical location is unequipped or preprovisioned. The validity status is Invalid.

If the layer rate is L3_Optical_Channel (which requires a DWDM CTP), CTM uses the naming convention defined in the TMF for the CTP.

The CTP format description differs depending on whether it is related to the DWDM. L3_Optical_Channel is the only layer rate that requires it. If the list of requested layer rates is composed of one element only equal to LR_Optical_Channel, the CTP is considered a DWDM CTP.

Parameters

Name	Type	Input/Output	Description
client	nmsSession::NmsSession_I	In	Reference to NMSSession. It cannot be null and it must be registered with CTM using emsSessionFactory::EmsSessionFactory_I::getEmsSession. CTM notifies this client about the completion or failure using methods defined in 3.12.1 historyPMDataCompleted, page 3-208 and 3.12.2 historyPMDataFailed, page 3-208.
pmTPSelectList	PMTPSelectList_T	In	This parameter specifies which historical PM data to return. This list cannot be empty.
pmParameters	PMParameterNameList_T	In	This parameter specifies which PM parameters within the scope of the pmTPSelectList shall be contained in the file. An empty list means to store all supported parameters. The returned parameters are best effort; that is, among the parameters specified, only the subset of supported parameters will be stored.
startTime	globaldefs::Time_T	In	Specifies the start of the time window for collection (included).
endTime	globaldefs::Time_T	In	Specifies the end of the time window for collection (excluded).

CTM support for PMTPSelect_T structure for this method is as follows:

- globaldefs::NamingAttributes_T name

  The name of the object to which this selection applies. CTM supports only ManagedElement, PTP, or CTP name in this field. Note that ManagedElement must be in CTM's management domain. For example, if you change the name of the ME but you pass the old name to this call, no data is written to the file.

- transmissionParameters::LayerRateList_T layerRateList

  For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- PMLocationList_T pMLocationList

  CTM supports only PML_NEAR_END_RX and PML_FAR_END_RX.

- GranularityList_T granularityList

  CTM supports 15-minute (15min) and 24-hour (24h) PM data. If this list is empty, data for all granularities is reported.

CTM supports the PM parameter values listed in Table 3-5, Table 3-6, Table 3-7, Table 3-9, Table 3-10, and Table 3-11.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when startTime or endTime are not valid, client is null, or
pmTPSelectList is empty.

EXCPT_UNABLE_TO_COMPY – Raised when more than one request is sent from the same NMS
client.
```

Compliance

CTM-defined.

3.13.6 getCTMValidHistoryPMData

Synopsis

```
Public void getCTMValidHistoryPMData (
             in nmsSession::NmsSession_I client,
             in PMTPSelectList_T pmTPSelectList,
             in PMParameterNameList_T pmParameters,
             in globaldefs::Time_T startTime,
             in globaldefs::Time_T endTime)
             raises(globaldefs::ProcessingFailureException);
```

Description

This interface performs exactly like the performance::PerformanceManagementMgr_I:: getCTMHistoryPMData interface, but filters out invalid entries.

Compliance

CTM-defined.

3.13.7 deleteHistoryPMDataFile

Synopsis

```
    public void deleteHistoryPMDataFile(
        in string fileName)
raises(globaldefs::ProcessingFailureException);
```

Description

The NMS can delete the PM data file by invoking this method explicitly. In this case, CTM does not wait for six hours to delete the file. This frees up valuable disk space on the CTM server. It is highly recommended that the NMS invoke this method as soon as it retrieves the PM data file from CTM.

Parameters

Name	Type	Input/Output	Description
filename	string	In	Name of the file that was created by CTM in response to the getCTMHistoryPMData call made by the NMS. This name must be the absolute path on the CTM server.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.13.8 getTPHistoryPMData

Synopsis

```
void getTPHistoryPMData(in PMTPSelectList_T pmTPSelectList,
      in PMParameterNameList_T pmParameters,
      in globaldefs::Time_T startTime,
      in globaldefs::Time_T endTime,
      in unsigned long how_many,
      out PMDataList_T pmDataList,
      out PMDataIterator_I pmIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to request a filtered set (scoped by the input parameter pmParameters) of the historical PM data for a list of TP measurement points. Within the request for each measurement point, granularity (15 min, 24 hr, NA) and location (nearEnd, farEnd, bidirectional) can be specified. The operation mode is best effort. CTM only supports 24 buckets (6 hours) for 15 min PM. If the NMS requests older historical data, CTM returns an EXCPT_NOT_IMPLEMENTED exception. CTM also returns this exception if the NMS requests historical data for 24-hour PM.

This interface uses an iterator so that the NMS can handle a large number of objects.

For a list of supported layer rates and PM parameters, with the exception of ONS 15305 and ONS 15305 CTC, see 3.13.5 getCTMHistoryPMData, page 3-268. The list for ONS 15305 and ONS 15305 CTC is shown in this section.

If the layer rate is L3_Optical_Channel (which requires a DWDM CTP), CTM uses the naming convention defined in the TMF for the CTP.

The CTP format description differs depending on whether it is related to the DWDM. L3_Optical_Channel is the only layer rate that requires it. If the list of requested layer rates is composed of one element only equal to LR_Optical_Channel, the CTP is considered a DWDM CTP.

Parameters

Name	Type	Input/Output	Description
pmTPSelectList	PMTPSelectList_T	In	Specifies which historical PM data to return. This list cannot be empty.
pmParameters	PMParameterNameList_T	In	Specifies which PM parameters within the scope of the pmTPSelectList are contained in the file. An empty list means to store all supported parameters. The returned parameters are best effort; that is, among the parameters specified, only the subset of supported parameters is stored.
startTime	globaldefs:Time_T	In	Specifies the start of the time window for collection (included).
endTime	globaldefs:Time_T	In	Specifies the end of the time window for collection (excluded).

Name	Type	Input/Output	Description
how_many	unsigned long	In	Maximum amount of PM data to return in the first batch.
pmDataList	PMDataList_T	Out	First batch of PM data.
pmIt	PMDataIterator	Out	Iterator used to retrieve the remaining PM data.

CTM support for PMData_T structure is as follows:

- globaldefs::NamingAttributes_T tpName

  The name of the termination point.

- transmissionParameters::LayerRate_T layerRate

  Layer of the collected PM data.

  CTM supports the following layer rates for ONS 15305 and ONS 15305 CTC. For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

- Granularity_T granularity

  Identifies whether this is 15-minute or 24-hour data. Only 15 min is supported.

CTM supports the pmParamater values listed in Table 3-11 for ONS 15305 and ONS 15305 CTC. For values for other NEs, see 3.13.5 getCTMHistoryPMData, page 3-268.

Table 3-11 ONS 15305 and ONS 15305 CTC pmParameter Values

ONS 15305 Layer Rate	pmParameterName	Location	Interval
LR_Line_OC12_STS12_and_MS_STM4	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr
LR_Line_OC3_STS3_and_MS_STM1	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr
LR_Line_OC48_STS48_and_MS_STM16	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr
LR_Low_Order_TU3_VC3	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr
LR_Section_OC12_STS12_and_RS_STM4	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX	15 min 24 hr
LR_Section_OC3_STS3_and_RS_STM1	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX	15 min 24 hr

Table 3-11 ONS 15305 and ONS 15305 CTC pmParameter Values (continued)

ONS 15305 Layer Rate	pmParameterName	Location	Interval
LR_Section_OC48_STS48_and_RS_STM16	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX	15 min 24 hr
LR_STS3c_and_AU4_VC4	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr
LR_VT2_and_TU12_VC12	PMP_BBE PMP_ES PMP_SES PMP_UAS	PML_NEAR_END_RX PML_FAR_END_RX	15 min 24 hr

- globaldefs::Time_T retrievalTime

  ME time at which PM measurement was obtained from the ME.
- PMMeasurementList_T pmMeasurementList:

  PM values.
- CTM support for PMMeasurement_T structure is as follows:
  - PMParameterName_T pmParameterName

    Name of the PM measure.
  - PMLocation_T pmLocation

    CTM supports PML_NEAR_END_RX and PML_FAR_END_RX.
  - float value

    Value of parameter.
  - string unit

    Not supported.
  - string intervalStatus

    CTM supports the following string values:

    Valid—The PM data value is valid.

    Invalid—The PM data value is invalid or unavailable.

Throws

```
EXCPT_NOT_IMPLEMENTED - Raised when the NMS queries historical data older than 6 hours for
15-minute buckets or 24-hour PM.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when any input parameter is not well formed.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

CTM-defined.

3.13.9 getHoldingTime

Synopsis

```
void getHoldingTime(out HoldingTime_T holdingTime)
    raises(globaldefs::ProcessingFailureException);
```

Description

The NMS requests from the EMS the number of hours that it stores 15-minute and 24-hour PM data records. This interface reports the minimum time. If the EMS does not store PM data, this method reports the time supported by the NE (the minimum time of all managed NEs).

Note CTM gives you the option to configure the holding time through the CTM GUI client.

Parameters

Name	Type	Input/Output	Description
holdingTime	HoldingTime_T	Out	Time period (in hours) within which 15-minute and 24-hour PM data records can be retrieved.

CTM support for the HoldingTime_T structure is as follows:

- Short storeTime24hr—Minimum store time, in hours, for 24-hour data. CTM reports 0 if PM data is held indefinitely.
- Short storeTime15min—Minimum store time, in hours, for 15-minute data. CTM reports 0 if PM data is held indefinitely.

Throws

```
EXCPT_NOT_IMPLEMENTED - Raised if the EMS is unable to support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NE_COMM_LOSS - Raised if communication to an NE is lost and the holding time cannot
be retrieved.
```

Compliance

TMF-defined.

3.13.10 getAllCurrentPMData

Synopsis

```
void getAllCurrentPMData(in PMTPSelectList_T pmTPSelectList,
      in PMParameterNameList_T pmParameters,
      in unsigned long how_many,
      out PMDataList_T pmDataList,
      out PMDataIterator_I pmIt)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables an NMS to request a filtered set (scoped by the input parameter pmParameters) of the current PM data for a list of TP measurement points. Within the request for each measurement point, granularity (15 min, 24 hr, N/A) and location (nearEnd, farEnd, bidirectional) can be specified. The operation mode is best effort.

Current data is generally marked as incomplete, as the current bin is not completed. If no PM data is available as specified, an empty list is returned.

This interface uses an iterator to allow the NMS to handle a large number of objects.

For a list of supported layer rates and PM parameters, see 3.13.5 getCTMHistoryPMData, page 3-268.

If the layer rate is L3_Optical_Channel (which requires a DWDM CTP), CTM uses the naming convention defined in the TMF for the CTP.

The CTP format description differs depending on whether it is related to the DWDM. L3_Optical_Channel is the only layer rate that requires it. If the list of requested layer rates is composed of one element only equal to LR_Optical_Channel, the CTP is considered a DWDM CTP.

The getAllCurrentPMData interface returns values with a 15-minute granularity if the requested granularity is 24 hour. For a list of supported layer rates, see 2.2.1 Layer Rate, page 2-23.

Parameters

Name	Type	Input/Output	Description
pmTPSelectList	PMTPSelectList_T	In	Specifies which real-time PM data to return. This list cannot be empty.
pmParameters	PMParameterNameList_T	In	Specifies which PM parameters within the scope of the pmTPSelectList are contained in the file. An empty list means to store all supported parameters. The returned parameters are best effort, meaning among the parameters specified, only the subset of supported parameters is stored.
how_many	unsigned long	In	Maximum number of PM data to return in the first batch.
pmDataList	PMDataList_T	Out	First batch of PM data.
pmIt	PMDataIterator_I	Out	Iterator used to retrieve the remaining PM data.

For detail of PMData_T structure, see 3.13.8 getTPHistoryPMData, page 3-271.

Throws

```
EXCPT_NOT_IMPLEMENTED - Raised when this operation is not supported by the EMS or not
supported for an ME type.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when any input parameter is not well formed.

EXCPT_NE_COMM_LOSS - Raised when communication to the NE is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

TMF-defined.

3.13.11 getMEPMcapabilities

Synopsis

```
void performance::PerformanceManagementMgr_I::getMEPMcapabilities  (  in
globaldefs::NamingAttributes_T  meName,
  in transmissionParameters::LayerRate_T  layerRate,
  out PMParameterList_T  pmParameterList
 )  raises (globaldefs::ProcessingFailureException)
```

Description

This interface allows the NMS to request from the EMS the set of PM parameters supported by a particular ME at a specific layer.

For a list of supported layer rates, see 3.13.5 getCTMHistoryPMData, page 3-268.

Parameters

Name	Type	Input/Output	Description
meName	NamingAttributes_T	In	Name of the managed element.
LayerRate	LayerRate_T	In	Layer rate at which PM parameters must be displayed.
pmParameterList	PMParameterList_T	Out	List of PMParameter_T, which returns the PM parameter name of the PM measure qualified by its location.

Throws

```
EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when meName does not reference a managed element object or
layerRate contains undefined rates.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.14 protection::ProtectionMgr_I

A handle to an instance of this interface is gained through the getManager operation in 3.4 EmsSession::EmsSession_I, page 3-50. This interface is inherited from 3.2 common::Common_I, page 3-18.

```
interface ProtectionMgr_I: common::Common_I
```

It also implements the getCapabilities method. Operations described in the following subsections are returned to the NMS when it invokes getCapabilities.

CTM supports 1:1 equipment protection, 1:N equipment protection, 1+1 port protection, 2-fiber BLSR protection groups, and 4-fiber BLSR protection groups.

3.14.1 getAllProtectedTPNames

Synopsis

```
void getAllProtectedTPNames(
in globaldefs::NamingAttributes_T pgName,
      in unsigned long how_many,
      out globaldefs::NamingAttributesList_T nameList,
      out globaldefs::NamingAttributesIterator_I nameIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the list of PTPs that can carry protected traffic. The result is independent of the current switch status. You can use the CTM client GUI to create and name protection groups. If there are multiple protection groups with the same name, CTM returns the ProcessingFailureException UNABLE_TO_COMPLY.

Parameters

Name	Type	Input/Output	Description
pgName	globaldefs::NamingAttributes_T	In	Name of the protection group.
how_many	unsigned long	In	Maximum number of PTPs to return in the first batch.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of PTPs that can carry protected traffic. Protected PTPs are reported.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator used to retrieve the remaining PTPs that can carry protected traffic.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when pgName does not reference a protection group object.

EXCPT_ENTITY_NOT_FOUND - Raised when pgName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.14.2 getAllProtectionGroups

Synopsis

```
void getAllProtectionGroups(
in globaldefs::NamingAttributes_T meName,
      in unsigned long how_many,
      out ProtectionGroupList_T pgList,
      out ProtectionGroupIterator_I pgpIt)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface is used by the client to discover all of the protection groups currently in operation for the ME. Note that CTM does not have any protection groups for path protection.

The supported protection group types are PGT_MSP_1_FOR_N, PGT_MSP_1_PLUS_1, PGT_2_FIBER_BLSR, and PGT_4_FIBER_BLSR.

The following table lists default names for the protection groups when created through CTC.

PGT_MSP_1_FOR_1	Slot *n* (Equipment Native Name), where *n* represents the slot number of the protecting equipment. For example, slot 3 (DS3IN).
PGT_MSP_1_FOR_N	Slot *n* (Equipment Native Name), where *n* represents the slot number of the protecting equipment. For example, slot 3 (DS3XM).
PGT_MSP_1_PLUS_1	Slot *n* (Equipment Native Name), port *m*, where *n* represents the slot number and *m* represents the protecting port number. For example, slot 5 (OC3), port 1.
PGT_2_FIBER_BLSR	No defaults; the name is a user-specified ring ID from 1 to 9999.
PGT_4_FIBER_BLSR	No defaults; the name is a user-specified ring ID from 1 to 9999.

If the PGP configuration is 4-fiber BLSR, there are three protection groups reported for this configuration: two PGT_1_FOR_N component groups and one PGT_4_FIBER_BLSR group.

The ONS 15600 SONET and ONS 15600 SDH support only PGT_MSP_1_PLUS_1 protection group.

Parameters

Name	Type	Input/Output	Description
meName	Globaldefs::NamingAttributes_T	In	Name of the ME for which the request is made.
how_many	unsigned long	In	Maximum number of protection groups to report in the first batch.
pgList	ProtectionGroupList_T	Out	First batch of protection groups.
PgIt	ProtectionGroupIterator_I	Out	Iterator used to access the remaining protection groups (PGs), if any.

CTM support for ProtectionGroup_T structure is as follows:

- globaldefs::NamingAttributes_T name

  Supported. This field contains three tuples. The first tuple contains the EMS name; the second tuple contains the ME name; and the last tuple contains the nativeEMSName of this protection group.

- string userLabel

  Not supported.

- string nativeEMSName

  Supported. The name is reported by the NE. It is Ring Id for 2_FIBER_BLSR and 4_FIBER_BLSR. It is <Ring Id>-EAST or <Ring Id>-WEST for the two component groups of 4-fiber BLSR PGP.

- string owner

  Not supported.

- ProtectionGroupType_T protectionGroupType

  Supported.

- ProtectionSchemeState_T protectionSchemeState

  Not supported.
- ReversionMode_T reversionMode

  Supported.
- transmissionParameters::LayerRate_T rate

  Line rate of the PTPs of the protection group.
- globaldefs::NamingAttributesList_T pgpTPList

  Supported.
- globaldefs::NVSList_T pgpParameters

  CTM only supports wtrTime, the values of wtrTime are in seconds.
- globaldefs::NVSList_T additionalInfo

  CTM supports BIDIRECTIONAL for 1_PLUS_1 protection group. The values are true and false.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when meName does not reference a managed element object.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

TMF-defined.

3.14.3 getProtectionGroup

Synopsis

```
void getProtectionGroup(
in globaldefs::NamingAttributes_T pgName,
      out ProtectionGroup_T protectionGroup)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the current status of a protection group. This service is needed so that even if a filter is established, the NMS can query the status of a protection group. If there is more than one protection group with the same name, CTM returns the ProcessingFailureException UNABLE_TO_COMPLY.

Parameters

Name	Type	Input/Output	Description
pgName	globaldefs::NamingAttributes_T	In	Name of the protection group that the client is interested in.
protectiongroup	ProtectionGroup_T	Out	Returned protection group.

For details about the ProtectionGroup_T structure, see 3.14.2 getAllProtectionGroups, page 3-277.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when protectionGroupName does not reference a protection
group.

EXCPT_ENTITY_NOT_FOUND - Raised when meName references an object that does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.14.4 performProtectionCommand

Synopsis

```
void performProtectionCommand(
in ProtectionCommand_T protectionCommand,
      in globaldefs::NamingAttributes_T reliableSinkCtpOrGroupName,
      in globaldefs::NamingAttributes_T fromTp,
      in globaldefs::NamingAttributes_T toTp,
      out SwitchData_T switchData)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface executes a protection switch. The protection switch can be performed through a protection switch command on a protection group. CTM requests the EMS to move the traffic received from the fromTP to the toTP. The same command is used to clear all existing commands. If there is more than one protection group with the same name, CTM returns the ProcessingFailureException UNABLE_TO_COMPLY.

A protection switch command is used to change the flow of traffic from working to protect or from protect to working entity. All entities that are represented as working entities before the switch are still represented as working entities after the switch. Similarly, all entities that are represented as protect entities before the switch are still represented as protect entities after the switch.

For 4-fiber SONET/SDH SPrings, the span switch is performed on the PGT_MSP_1_FOR_N component protection group, and the ring switch on the PGT_4_FIBER_BLSR group. For example, if the switch is on the east span, the PGP name is *ring-ID*-EAST.

For 2-fiber ring switch, it can be either E -> W or W -> E.

For 4-fiber ring switch, it can be Ew (east working) -> Wp (west protecting), Ep -> Ww, Ww -> Ep, and Wp -> Ew.

For 4-fiber span switch, it can be Ew -> Ep, Ep ->Ew, Ww -> Wp, and Wp -> Ww.

CTM validates whether or not fromTp and toTp are part of same protection group. If this validation fails then it throws ProcessingFailureException with EXCPT_INVALID_INPUT. CTM always performs the command on the fromTp for PC_LOCKOUT and PC_EXERCISE. CTM also verifies if it is possible to perform the command; if not, it throws EXCPT_UNABLE_TO_COMPLY.

Table 3-12 Protection Command Protection Groups

Protection Group	Protection Command Supported by CTM
PGT_2_FIBER_BLSR	PC_CLEAR PC_LOCKOUT PC_FORCED_SWITCH PC_MANUAL_SWITCH PC_EXERCISE
PGT_4_FIBER_BLSR **Note** Protection command applies to both span and ring components.	PC_CLEAR PC_LOCKOUT PC_FORCED_SWITCH PC_MANUAL_SWITCH PC_EXERCISE
PGT_MSP_1_PLUS_1	PC_CLEAR, PC_LOCKOUT, PC_FORCED_SWITCH, PC_MANUAL_SWITCH PC_UNLOCK
PGT_MSP_1_FOR_N	PC_CLEAR PC_LOCKOUT PC_FORCED_SWITCH PC_MANUAL_SWITCH PC_UNLOCK
PGT_MSP_1_FOR_1	PC_CLEAR PC_LOCKOUT PC_FORCED_SWITCH PC_MANUAL_SWITCH PC_UNLOCK

Parameters

Name	Type	Input/Output	Description
protectionCommand	ProtectionCommand_T	In	Command to perform.
reliableSinkCtpOrGroupName	globaldefs::NamingAttributes_T	In	CTM supports only protection group name.

Name	Type	Input/Output	Description
fromTp	globaldefs::NamingAttributes_T	In	Present source of the traffic.
toTp	globaldefs::NamingAttributes_T	In	Requested source of the traffic after the command.
switchData	SwitchData_T	Out	Protection switch status of the toTp provided after the execution of the command.

CTM support for SwitchData _T structure is as follows:

- ProtectionType_T protectionType

  Supported. CTM always reports PT_MSP_APS.
- SwitchReason_T switchReason

  Supported. CTM always reports SR_MANUAL.
- transmissionParameters::LayerRate_T layerRate

  The layer that this switch is relevant to.
- globaldefs::NamingAttributes_T groupName

  Identifies the protection group for which protection switch status is being reported.
- globaldefs::NamingAttributes_T protectedTP

  This is the TP that is protected.
  - For a 2-fiber BLSR, this will be from TP.
  - For a 1:1, this is always the working TP.
  - For a 1:N MSP, this is always the working TP.
  - For a 1+1 MSP, this is always the working TP.
  - For a 4-fiber BLSR, this is the working TP.
- globaldefs::NamingAttributes_T switchToTP

  CTM reports toTp.
- globaldefs::NVSList_T additionalInfo

  Not supported.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when reliableSinkCtpOrGroupName, fromTp, or toTp references
objects of the correct type, but that are invalid in the context of this operation.

EXCPT_ENTITY_NOT_FOUND - Raised when reliableSinkCtpOrGroupName references a CTP or PG
object that does not exist, or when fromTp or toTp references a CTP object that does not
exist.

EXCPT_UNABLE_TO_COMPLY - Raised if the EMS is unable to perform the operation.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

TMF-defined.

3.14.5 createProtectionGroup

Synopsis

```
void createProtectionGroup(
in PGPCreateData_T pgpCreateData,
      out ProtectionGroup_THolder createdPGP)
    raises(globaldefs::ProcessingFailureException);
```

Description

This interface enables the NMS to permanently create a protection group in an ME. The successful result of this operation is the creation of the protection group object. The resulting object is returned. CTM generates an object creation event upon successful creation of the protection group.

CTM supports only the creation of 1_PLUS_1, 1_FOR_N, 1_FOR_1 (a specific case where the cards are placed in adjacent slots), 2-fiber BLSR, 4-fiber BLSR, SPLITTER, and L2_1_PLUS protection groups.

Note CTM enables only the creation of 2-fiber and 4-fiber protection groups at the managed element level and not at the network level.

For the ONS 15305 R3.0.x, the name of the 1+1 MSP protection group is assigned by the NE itself. The pgpName attribute that is passed as input is ignored. The object creation event reports the protection group name that is set by the NE.

Parameters

Name	Type	Input/Output	Description
pgpCreateData	PGPCreateData_T	In	Data that describes the protection group to create.
createdPGP	ProtectionGroup_T	Out	Resulting created protection group.

CTM support for PGPCreateData_T structure is as follows:

- PGPCreateData_T structure for 1+1 and 1:N protection group creation.

Type	Parameter Name	1+1	1:N	1:1
NamingAttributes_T	pgpName	Protection group name	Protection group name	Protection group name
string	userlabel	N/A	N/A	N/A
boolean	forceUniqueness	False	False	False
string	owner	N/A	N/A	N/A
ProtectionGroupType_T	pgpType	PGT_MSP_1_PLUS_1	PGT_MSP_1_FOR_N	PGT_MSP_1_FOR_1

Type	Parameter Name	1+1	1:N	1:1
ReversionMode_T	reversionMode	Could be revertive or nonrevertive ReversionMode_T. RM_REVERTIVE or ReversionMode_T. RM_NON_REVERTIVE	ReversionMode_T. RM_REVERTIVE **Note** 1:N is always revertive.	Could be revertive or nonrevertive ReversionMode_T. RM_REVERTIVE or ReversionMode_T. RM_NON_REVERTIVE
NamingAttributesList_T	workingEquipmentOrTPList	Working PTP name	Name of working equipment	Name of working equipment
NamingAttributesList_T	protectEquipmentOrTPList	Protect PTP name	Name of protect equipment	Name of protect equipment
NVList_T	pgpParameters	Name: WTR_TIME (reversion time) Value: Time in seconds	Name: WTR_TIME (reversion time) Value: Time in seconds	Name: WTR_TIME (reversion time) Value: Time in seconds
NVList_T	additionalInfo	Name: BIDIRECTIONAL Value: true or false	N/A	N/A

- PGPCreateData_T structure for 2-fiber/4-fiber BLSR ring creation.

Type	Parameter Name	2-Fiber BLSR	4-Fiber BLSR
NamingAttributes_T	pgpName	Ring name	Ring name
string	userlabel	N/A	N/A
boolean	forceUniqueness	False	False
string	owner	N/A	N/A
ProtectionGroupType_T	pgpType	PGT_2_FIBER_BLSR	PGT_4_FIBER_BLSR
ReversionMode_T	reversionMode	Always revertive. ReversionMode_T. RM_REVERTIVE	ReversionMode_T. RM_REVERTIVE
NamingAttributesList_T	workingEquipmentOrTPList	First element in the list contains the west-line PTP name Second element in the list contains the east-line PTP name	First element in the list contains the west-line PTP name Second element in the list contains the east-line PTP name
NamingAttributesList_T	protectEquipmentOrTPList	N/A. Must be an array of 0 length	First element in the list contains the west-line protect PTP Second element contains the east-line protect PTP

Type	Parameter Name	2-Fiber BLSR	4-Fiber BLSR
NVList_T	pgpParameters	Name: WTR_TIME (reversion time) Value: Time in seconds Name: SPRINGNodeId (Node ID) Value: 0–31	Name: WTR_TIME (reversion time) Value: Time in seconds Name: SPRINGNodeId (Node ID) Value: 0–31
NVList_T	additionalInfo	N/A **Note** If there is no additional info, an array of zero length is passed as the input parameter. The list cannot be null.	Name: SPAN_REVERSION Value: 0–600 seconds (in 30-second increments) If not specified, CTM uses the default value of 300 seconds

- PGPCreateData_T structure for SPLITTER and L2_1_PLUS protection.

SPLITTER	L2 -1_PLUS_1
Protection group name	Protection group name
N/A	N/A
False	False
N/A	N/A
PGT_SPLITTER	PGT_L2_1_PLUS_1
Could be revertive or non-revertive ReversionMode_T. RM_REVERTIVE or ReversionMode_T. RM_NON_REVERTIVE	Could be revertive or non-revertive ReversionMode_T. RM_REVERTIVE or ReversionMode_T. RM_NON_REVERTIVE
Working PTP name	Working PTP name
Protect PTP name	Protect PTP name
Name: WTR_TIME (reversion time) Value: Time in seconds	Name: WTR_TIME (reversion time) Value: Time in seconds
N/A	N/A

The following enum is used to define the various protection group types:

```
enum ProtectionGroup_T {
   PGT_MSP_1_PLUS_1,
   PGT_MSP_1_FOR_1,
   PGT_MSP_1_FOR_N,
   PGT_2_FIBER_BLSR,
   PGT_4_FIBER_BLSR
   PGT_SPLITTER
   PGT_L2_1_PLUS_1
   }
```

The following enum is used to define the reversion mode:

```
enum ReversionMode_T {
RM_UNKNOWN,
```

```
RM_REVERTIVE,
RM_NON_REVERTIVE
}
```

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS cannot support this service.

EXCPT_INVALID_INPUT - Raised when equipment or TP name does not reference an equipment or
TP object.

EXCPT_ENTITY_NOT_FOUND - Raised when equipment or TP name references an object that does
not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the protection group cannot be created.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.
```

Compliance

CTM-defined.

3.14.6 deleteProtectionGroup

Synopsis

```
void deleteProtectionGroup(
    in  globaldefs::NamingAttributes_T pgpName)
raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows the NMS to permanently delete a protection group from the ME. The successful result of this operation is the potential deletion of the protection group object. CTM generates an object deletion event on successful deletion of the protection group. CTM supports deletion of all protection group types. If there is more than one protection group with the same name, CTM returns an UNABLE_TO_COMPLY processing failure exception.

Parameters

Name	Type	Input/Output	Description
pgpName	globaldefs::NamingAttributes_T	In	Name of the protection group to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when pgpName does not reference a protection group object.

EXCPT_ENTITY_NOT_FOUND - Raised when pgpName references an object that does not exist.

EXCPT_UNABLE_TO_COMPLY - Raised when the protection group cannot be deleted from the NE.
```

```
EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.14.7 getAssociatedProtectionGroup

Synopsis

```
void getAssociatedProtectionGroup(
       in globaldefs::NamingAttributes_T ptpName,
       out protection::ProtectionGroup_T protectionGrp)
   raises(globaldefs::ProcessingFailureException);
```

Description

This interface retrieves the protection group using the specified PTP name. The PTP name specified can be either the working PTP or the protect PTP.

Parameters

Name	Type	Input/Output	Description
ptpName	globaldefs::NamingAttributes_T	In	Name of the PTP to verify an association to any protection group.
protectiongroup	ProtectionGroup_T	Out	Returned protection group.

For details about the ProtectionGroup_T structure, see 3.14.2 getAllProtectionGroups, page 3-277.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when ptpName does not reference a PTP.

EXCPT_ENTITY_NOT_FOUND - Raised when the PTP name references an object that does not
exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.15 session::Session_I

The Session_I interface provides capabilities for managing the client-server connection. It enables either a client or a server to detect communication loss with the associated party. There are two Session_I objects for each communication session between an NMS and an EMS. One Session_I object is maintained on the NMS and the other Session_I object is maintained on the EMS. The Session_I object maintained on the EMS is actually an EmsSession_I, while the Session_I object maintained on the NMS is actually an NmsSession_I (both inherit from Session_I).

Each Session_I object is responsible for pinging the other Session_I object periodically to detect communication failures. CTM invokes this method on NmsSession_I every five minutes. If this call raises an exception, CTM assumes the session is no longer connected and deletes all resources allocated to it.

Attributes

```
readonly attribute Session_I associatedSession;
```

This attribute contains a reference to the Session_I on the NMS/EMS associated with the object. This attribute can be checked to verify the NmsSession_I/EmsSession_I association is still valid (particularly in case of communication failures).

3.15.1 endSession

Synopsis

```
oneway void endSession();
```

Description

This interface enables a controlled disconnect between parties.

Parameters

None.

Throws

None.

Compliance

TMF-defined.

3.15.2 ping

Synopsis

```
void ping();
```

Description

This interface enables the detection of communication loss. CTM pings the NMS every five minutes.

Parameters

None.

Throws

None.

Compliance

TMF-defined.

3.16 mtnmVersion::Version_I

3.16.1 getVersion

Synopsis

```
string getVersion();
```

Description

This interface returns the version of the IDL that the corresponding EMS object supports.

The format of the returned string is as follows:

Release.Major[.Minor], where *Release*, *Major*, and *Minor* are digit strings.

For example:

- *2.1* indicates release 2 and major release 1
- *1.3* indicates release 1 and major release 3

Note *x.y* has the same meaning as *x.y.0.* The minor digit is used for fixing bugs in the major release. For example, *1.2.1* is a minor release on *1.2*. Any changes to IDL files must be approved by the Specification Authority. For TMF 814 version 2.0, the version is *2.0*.

Parameters

None.

Return Value

```
string
```

Throws

None.

Compliance

TMF-defined.

3.17 Transmission Descriptor Interfaces

```
enum ServiceCategory_T {
     SC_CBR,
     SC_VBRRT,
     SC_VBRNRT,
     SC_ABR,
     SC_UBR,
     SC_GFR,
     SC_NA
    };


    typedef globaldefs::NVSList_T TrafficParameterList_T;
```

```
struct TransmissionDescriptor_T {
  globaldefs::NamingAttributes_T name;
  string userLabel;
  string nativeEMSName;
  string owner;
  transmissionParameters::LayeredParameterList_T transmissionParams;
  globaldefs::NVSList_T additionalTPInfo;
  globaldefs::NamingAttributes_T containingTMDName;
  string externalRepresentationReference;
  globaldefs::NVSList_T additionalInfo;
};


struct TMDCreateData_T {
 string userLabel;
 boolean forceUniqueness;
 string owner;
 transmissionParameters::LayeredParameterList_T transmissionParams;
 globaldefs::NVSList_T additionalTPInfo;
 globaldefs::NamingAttributes_T containingTMDName;
 string externalRepresentationReference;
 globaldefs::NVSList_T additionalCreationInfo;
};


typedef sequence<TransmissionDescriptor_T> TransmissionDescriptorList_T;
```

3.17.1 getAllTransmissionDescriptors

Synopsis

```
void getAllTransmissionDescriptors(
          in unsigned long how_many,
          out TransmissionDescriptorList_T transmissionDescList,
          out TransmissionDescriptorIterator_I transmissionDescIt)
         raises (globaldefs::ProcessingFailureException);
```

Description

This interface allows an NMS to request all of the transmissionDescriptors that are under the control of the transmissionDescriptorMgr_I. To allow the NMS to handle a large number of objects, this operation uses an iterator.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Number of iterators to return in transmissionDescList.
transmissionDescList	transmissionDescriptor::TransmissionDescriptorList	Out	First batch of iterators.
transmissionDescIt	transmissionDescriptor::TransmissionDescriptorIterator_I	Out	Iterator used to access the remaining transmission descriptors.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

None.

3.17.2 getAllTransmissionDescriptorsNames

Synopsis

```
void getAllTransmissionDescriptorNames(
          in unsigned long how_many,
          out globaldefs::NamingAttributesList_T nameList,
          out globaldefs::NamingAttributesIterator_I nameIt)
         raises(globaldefs::ProcessingFailureException);
```

Description

This interface returns the transmission descriptor structure for the given tmdName. The transmission descriptor structure contains an NVSList of traffic parameters. The traffic parameters returned are the parameters in place on the actual transmission descriptor. If there are no traffic parameters, the NVSList is empty.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
how_many	unsigned long	In	Number of iterators to return in nameList.
nameList	globaldefs::NamingAttributesList_T	Out	First batch of iterators.
nameIt	globaldefs::NamingAttributesIterator_I	Out	Iterator used to retrieve the remaining iterator names.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

None.

3.17.3 getTransmissionDescriptor

Synopsis

```
void getTransmissionDescriptor(
          in globaldefs::NamingAttributes_T tmdName,
          out TransmissionDescriptor_T tmd)
         raises (globaldefs::ProcessingFailureException);
```

Description

This interface returns the transmission descriptor structure for the given tmdName. The transmission descriptor structure contains an NVSList of traffic parameters. The traffic parameters returned are the parameters in place on the actual transmission descriptor. If there are no traffic parameters, the NVSList is empty.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
tmdName	globaldefs::NamingAttributes_T	In	Name of the transmission descriptor.
tmd	TransmissionDescriptor_T	Out	Returned transmission descriptor.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when tmdName does not reference a transmission descriptor
object.

EXCPT_ENTITY_NOT_FOUND - Raised when tmdName references an object that does not exist.
```

Compliance

None.

3.17.4 getAssociatedTPs

Synopsis

```
void getAssociatedTPs(
          in globaldefs::NamingAttributes_T transmissionDescriptorName,
          in unsigned long how_many,
          out terminationPoint::TerminationPointList_T tpList,
          out terminationPoint::TerminationPointIterator_I tpIt)
         raises(globaldefs::ProcessingFailureException);
```

Description

This interface allows an NMS to request all of the TPs associated with the specified transmission descriptor. If there are no associated TPs, an empty list is returned.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
transmissionDescriptorName	globaldefs::NamingAttributes_T	In	Name of the transmission descriptor.
how_many	unsigned long	In	Maximum number of TPs to return in the first batch.
tpList	terminationPoint::TerminationPointList_T	Out	First batch of TPs.
tpIt	terminationPoint::TerminationPointIterator_I	Out	Iterator to access the remaining TPs.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when transmissionDescriptorName does not reference a
transmission descriptor object.

EXCPT_ENTITY_NOT_FOUND - Raised when transmissionDescriptorName references an object that
does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

None.

3.17.5 createTransmissionDescriptor

Synopsis

```
void createTransmissionDescriptor(
          in  TMDCreateData_T      newTMDCreateData,
          out TransmissionDescriptor_T newTransmissionDescriptor)
         raises (globaldefs::ProcessingFailureException);
```

Description

This interface is used to create a new transmission descriptor on the server. Data representing the new transmission descriptor is passed as input. The resulting transmission descriptor is returned.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
newTMDCreateData	TMDCreateData_T	In	Information about the transmission descriptor to create.
newTransmissionDescriptor	TransmissionDescriptor_T	Out	Result of the creation.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support creation of TMDs via this
interface.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised if newTMDCreateData contains invalid data.

EXCPT_USERLABEL_IN_USE - Raised when the user label uniqueness constraint is not met.

EXCPT_CAPACITY_EXCEEDED - Raised when the EMS reaches the maximum number of transmission
descriptors that it can support.
```

Compliance

None.

3.17.6 deleteTransmissionDescriptor

Synopsis

```
void deleteTransmissionDescriptor(
          in globaldefs::NamingAttributes_T transmissionDescriptorName)
          raises (globaldefs::ProcessingFailureException);
```

Description

This interface deletes a transmission descriptor from the server. If the service is called with the name of a nonexistent transmission descriptor, it succeeds.

This interface is introduced in CTM to comply with MTNM version 3.0. The current implementation returns a globaldefs::ProcessingFailureException stating that this API is not supported.

Parameters

Name	Type	Input/Output	Description
transmissionDescriptorName	globaldefs::NamingAttributes_T	In	Information about the transmission descriptor to delete.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support deletion of TMDs via this
interface.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

```
EXCPT_OBJECT_IN_USE - Raised if there are TPs that are using the transmission descriptor.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the transmission descriptor to delete does not exist.
```

Compliance

None.

3.18 Software and Data Manager Interface

The softwareAndDataManager interface is used to gain access to operations related to backing up ME configuration data.

A handle to an instance of this interface is gained via the emsSession::EmsSession_I::getManager() operation in Manager.

This interface is inherited from interface SoftwareAndDataMgr_I:common::Common_I.

The softwareAndDataManager interface also implements the getCapabilities method. When the NMS invokes the getCapabilities method, the operations described in the following sections are returned.

3.18.1 backupME

Synopsis

```
void backupME(
        in globaldefs::NamingAttributes_T managedElementName)
        raises(globaldefs::ProcessingFailureException)
```

Description

This asynchronous operation initiates a backup request on the specified ME. This creates a backup of the current configuration on the ME. The EMS sends the results of the ME backup operation via the notification service. The NMS can obtain the status of any backup operation on demand. An NT_BACKUP_STATUS event is sent to indicate the result of the backup operation.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which a backup of configuration data is required.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.
```

Compliance

TMF-defined.

3.18.2 getMEBackupStatus

Synopsis

```
void getMEBackupStatus (
        in globaldefs::NamingAttributes_T managedElementName,
        out BackupStatus_T  backupStatus)
        raises(globaldefs::ProcessingFailureException);
```

Description

The NMS uses this method to obtain the current backup status of an individual ME.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which the status of the backup configuration data is required.
backupStatus	BackupStatus_T	Out	Backup status of the ME.

BackupStatus_T is defined as:

Current_OperationStatus_T opStatus

string failureReason

This data structure identifies the status of the backup operation for a managed element. The failure reason is present if the operation status indicates a failure (in abort state).

- Current_OperationStatus_T opStatus—Indicates the current operational status of the backup. The structure of Current_OperationStatus_T is defined as:

  ```
  enum Current_OperationStatus_T { COS_Idle, COS_Pending, COS_InProgress, COS_Completed,
  COS_Aborted };
  ```

  where:

  - COS_Idle—The backup operation is waiting.
  - COS_Pending—The backup operation is queued.
  - COS_InProgress—The backup operation is running.
  - COS_Completed—The backup operation succeeded.
  - COS_Aborted—The backup operation was canceled.
- string failureReason—A free-form text string provided if the opStatus value is COS_Aborted to explain the reason for the abort. (For example, “Comms loss with NE.”)

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.
```

```
EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.
```

Compliance

TMF-defined.

3.18.3 abortMEBackup

Synopsis

```
void abortMEBackup (
        in globaldefs::NamingAttributes_T managedElementName)
        raises(globaldefs::ProcessingFailureException)
```

Description

This interface aborts a pending backup of ME configuration data. If the backup operation on the ME is in progress or if there are no pending backup operations, an exception is generated.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which a backup of configuration data is required.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.

EXCPT_NOT_IN_VALID_STATE - Raised when there is no pending or in-progress backup of
configuration data on the ME.
```

Compliance

TMF-defined.

3.18.4 getBackupList

Synopsis

```
void getBackupList(
        in globaldefs::NamingAttributesList_T managedElementNameList,
        in unsigned long how_many,
        out BackupIdList_T backupList,
        out BackupIdIterator_I backupIt)
        raises(globaldefs::ProcessingFailureException)
```

Description

The NMS invokes this method to determine what configuration data backups exist in the EMS for a specified list of MEs.

This interface returns a list of IDs of all configuration data backups for the specified MEs. If the list of MEs in the request is empty, the IDs of all configuration data backups are returned for all MEs.

To allow the NMS to handle a large number of objects, this operation uses an iterator. See the TMF document *Overview of Iterator Usage* for information on how iterators are used in this interface.

Parameters

Name	Type	Input/Output	Description
managedElementNameList	NamingAttributesList_T	In	List of names of MEs for which the configuration data IDs are returned. If one or more MEs do not exist, the operation stops.
how_many	long	In	Maximum number of configuration data backup IDs to report in the first batch.
backupList	BackupIdList_T	Out	First batch of configuration data backup IDs.
backupIt	BackupIdIterator_I	Out	Iterator to retrieve the remaining configuration data backup IDs.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when one or more MEs in the list do not exist.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the EMS reaches the maximum number of
iterators that it can support.
```

Compliance

TMF-defined.

3.18.5 activateSoftwareOnME

Synopsis

```
void activateSoftwareOnME(
        in globaldefs::NamingAttributes_T managedElementName)
        raises(globaldefs::ProcessingFailureException)
```

Description

The NMS uses this method to activate downloaded software on an ME.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T managedElementName	In	Name of the ME on which to activate software.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.
```

Compliance

CTM-defined.

3.18.6 revertSoftwareOnME

Synopsis

```
void revertSoftwareOnME(
        in globaldefs::NamingAttributes_T managedElementName)
        raises(globaldefs::ProcessingFailureException)
```

Description

The NMS uses this method to revert to the older software version on the specified managed element.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T managedElementName	In	Name of the ME on which to revert software back to the older version.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.
```

Compliance

CTM-defined.

3.18.7 restoreME

Synopsis

```
void restoreME(
        in globaldefs::NamingAttributes_T managedElementName,
        in string downloadFileName)
        raises(globaldefs::ProcessingFailureException)
```

Description

This asynchronous operation initiates a restore request on the specified ME. The ME configuration file to restore is obtained from the server.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which a restore of configuration data is required.
downloadFileName	String	In	Name of the file to restore.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.
```

Compliance

CTM-defined.

3.18.8 getMERestoreStatus

Synopsis

```
void getMERestoreStatus (
        in globaldefs::NamingAttributes_T managedElementName,
        out RestoreStatus_T  restoreStatus)
        raises(globaldefs::ProcessingFailureException)
```

Description

The NMS uses this method to obtain the current restore status of an individual ME.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which a restore status of configuration data is required.
restoreStatus	RestoreStatus_T	Out	Status of the configuration data restoration.

```
struct RestoreStatus_T
{
  Current_OperationStatus_T opStatus;
  string failureReason;
}
```

RestoreStatus_T is defined as:

- Current_OperationStatus_T opStatus—Indicates the current operational status of the restore.
- string failureReason—A free-form text string that is provided if the opStatus value is COS_Aborted to explain the reason for the abort. (For example, "Comms loss with NE.")

This data structure identifies the status of the restore operation for a managed element. The failure reason is present if the operation status indicates a failure (in abort state).

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.
```

Compliance

CTM-defined.

3.18.9 abortMERestore

Synopsis

```
void abortMERestore (
        in globaldefs::NamingAttributes_T managedElementName)
        raises(globaldefs::ProcessingFailureException)
```

Description

This interface aborts the restoration of pending ME configuration data. If the restore operation on the ME is in progress or if there are no pending restore operations, an exception is generated.

Parameters

Name	Type	Input/Output	Description
managedElementName	NamingAttributes_T	In	Name of the ME for which to stop the restoration of configuration data.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_NOT_IMPLEMENTED - Raised if the EMS does not support this service.

EXCPT_INVALID_INPUT - Raised when input parameters are syntactically incorrect.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.

EXCPT_NE_COMM_LOSS - Raised when communication to the ME is lost.

EXCPT_NOT_IN_VALID_STATE - Raised when there is no pending or in-progress restoration of
configuration data on the ME.
```

Compliance

CTM-defined.

3.18.10 getAllSupportedMESoftwareVersions

Synopsis

```
void getAllSupportedMESoftwareVersions(in globaldefs::NamingAttributes_T emsName ,
        in unsigned long how_many,
        out SupportedMESoftwareVersionList_T versionList,
        out SupportedMESoftwareVersionIterator_I versionIt)
        raises(globaldefs::ProcessingFailureException);
```

Description

The NMS uses this operation to retrieve a list of potential ME versions (or software versions) that can be managed by the EMS.

Parameters

Name	Type	Input/Output	Description
emsName	gloaldefs::NamingAttributes_T	In	Name of the EMS for which the supported ME software versions are retrieved.
how_many	unsigned long	In	Maximum number of software versions to report in the first batch.
versionList	SupportedMESoftwareVersionList_T	Out	First batch of supported software versions. (See the definition below.)
versionIt	SupportedMESoftwareVersionIterator_I	Out	Iterator to retrieve the remaining software versions. (See the definition below.)

```
struct SupportedMESoftwareVersion_T
  {
    string meModel;
    string swVersion;
   globaldefs:NVSList_T additionalInfo;
    };
```

where:

- String MEmodel—Model of the managed element (such as Cisco ONS 15454).

- String swVersion—Version of the ME software that the NMS can manage. This string is identical to the version attribute in the ManagedElement_T structure.
- globaldefs::NVSList_T additionalInfo—The EMS communicates to the NMS additional information that is not explicitly modeled.

To support retrieval in batches, a SupportedMESoftwareVersionIterator_I can be defined.

```
typedef sequence<SupportedMESoftwareVersion_T> SupportedMESoftwareVersionList_T;
interface SupportedMESoftwareVersionIterator_I
  {
     boolean next_n (in  unsigned long    how_many,
                     out SupportedMESoftwareVersionList_T neVersionList)
       raises (globaldefs::ProcessingFailureException);

     unsigned long getLength()
       raises (globaldefs::ProcessingFailureException);

     void destroy()
       raises (globaldefs::ProcessingFailureException);
  };
```

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_TOO_MANY_OPEN_ITERATORS - Raised when the maximum number of iterators reaches 128.
```

Compliance

CTM-defined.

3.18.11 downloadMESoftware

Synopsis

```
    void downloadMESoftware(
    in globaldefs::NamingAttributes_T managedElementName,
    in SWDownloadData_T swDownloadData)
     raises(globaldefs::ProcessingFailureException);
```

Description

This interface schedules a software download on a managed element.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T	In	Name of the ME on which to schedule a software download operation.
swDownloadData	SWDownloadData_T	In	Data needed to schedule a software download. (See the following definition.)

```
struct SWDownloadData_T
       {
        globaldefs::Time_T startTime;
        string userComment;
        FileSource source;
```

```
 string downLoadFileName;
 string remoteServerAddress;
 string ftpUserName;
 string ftpPassword;
};
```

The following table defines each parameter.

Parameters

Name	Type	Input/Output	Description
startTime	globaldefs::Time_T	In	Start time of the scheduled software download.
userComment	string	In	User comments.
source	FileSource	In	Image source (FTP, TFTP, server, and so on). (See the following definition.)
downLoadFileName	string	In	Name of the image to download.
remoteServerAddress	string	In	Address of the remote server.
ftpUserName	string	In	FTP username.
ftpPassword	string	In	FTP password.

The definition of the file source is:

```
enum FileSource{FROM_LOCAL, FROM_TFTP, FROM_FTP};
```

where:

- FROM_LOCAL—The file source is the same server (that is, the CTM server that you are currently logged into). The package file must be placed in the directory <CTM Server Installation Dir>/image/Cisco_ONS_<NE model Name>.
- FROM _TFTP—The file source is the remote server.
- FROM_FTP—The file source is the remote server.

Currently, CTM GateWay/CORBA does not support software download from the local machine.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised if the managed element does not exist.

EXCPT_INVALID_INPUT - Raised when invalid input is provided.

EXCPT_NE_COMM_LOSS - Raised when communication to the managed element is lost.
```

Compliance

CTM-defined.

3.18.12 addSupportedMESoftwareVersion

Synopsis

```
void addSupportedMESoftwareVersion(
```

```
in globaldefs::NamingAttributes_T emsName,
in SupportedMESoftwareVersion_T versionData)
raises (globaldefs::ProcessingFailureException)
```

Description

This interface allows an NMS to add a new software version to the list of supported ME software versions.

Parameters

Name	Type	Input/Output	Description
emsName	globaldefs::NamingAttributes_T	In	Name of the EMS on which to add a new supported ME software version.
versionData	SupportedMESoftwareVersion_T	In	Input details for the addition of a new software version.

```
struct SupportedMESoftwareVersion_T
  {
    string meModel;
    string swVersion;
   globaldefs:NVSList_T additionalInfo;
    };
```

The versionData contains all of the necessary fields specific to the EMS with the required information for the addition of a new supported version. All of the fields are supported and have the following descriptions:

- string meModel—The model string of the ME model type for which to add the software version. (For a complete set of strings for all MEs, see 3.9.44 addManagedElement, page 3-174.)
- string swVersion—The new software version to add.
- globaldefs:NVSList_T additionalInfo—The additionalInfo field consists of various parameters that must be specified to the EMS to add the software version. The structure of additionalInfo is as follows:

Name	Value	Description	Mandatory or Optional
ModelName	Any value (used for display purposes).	A version name used for display purposes. (This value usually indicates the model type and release.)	Mandatory.
OldSWVersion	A valid, existing software version.	The old software version of the ME model type for which to add the new software version.	Mandatory.

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INVALID_INPUT - Raised in case of invalid input. This could be specific to the EMS.

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.
```

Compliance

CTM-defined.

3.18.13 getSWDownloadStatus

Synopsis

```
void getSWDownloadStatus (
        in globaldefs::NamingAttributes_T managedElementName,
        out BackupStatus_T  backupStatus)
        raises(globaldefs::ProcessingFailureException)
```

Description

The NMS uses this method to obtain the current software download status of an individual ME.

Parameters

Name	Type	Input/Output	Description
managedElementName	globaldefs::NamingAttributes_T managedElementName	In	Name of the ME for which a software download status is required.
backupStatus	BackupStatus_T	Out	Status of the download.

```
struct BackupStatus_T
{
Current_OperationStatus_T opStatus;
string failureReason;
}
```

BackupStatus_T is defined as:

- Current_OperationStatus_T opStatus—Indicates the current operational status of the software download.
- string failureReason—A free-form text string that is provided if the opStatus value is COS_Aborted to explain the reason for the abort. (For example, “Comms loss with NE.”)

This data structure identifies the status of the software download operation for a managed element. The failure reason is present if the operation status indicates a failure (in abort state).

Throws

```
globaldefs::ProcessingFailureException

EXCPT_INTERNAL_ERROR - Raised in case of nonspecific EMS internal failure.

EXCPT_ENTITY_NOT_FOUND - Raised when the ME does not exist.
```

Compliance

CTM-defined.

APPENDIX A

Notifications, Fault Management, and Probable Cause Mapping

This appendix includes information on CTM GateWay/CORBA's notifications, fault management, and probable cause mapping. This appendix includes the following sections:

- A.1 Overview of Notifications, page A-1
- A.2 Resolving the Notification Service, page A-2
- A.3 Tips to Improve Notification Throughput, page A-2
- A.4 Supported Events, page A-3
- A.5 Probable Cause Mapping, page A-12

A.1 Overview of Notifications

The notification service forwards all fault notifications received from managed NEs to the network management system (NMS). The notification service also forwards object creation, object deletion, attribute change, protection switch, and threshold crossing alert (TCA) events. TMF 814 defines the structure and format of attributes for all events and alarms.

The CTM server and the Operations Support System (OSS) host must have Domain Name System (DNS) enabled in order to receive notifications. OSS clients running behind a firewall must configure an IIOP listening port to enable the TCP connection with the notification server. CTM must configure the notification service listening port number to allow OSS clients to communicate with the notification service.

CTM creates the notification channel with the QoS parameters shown in the following table.

Table A-1 QoS Parameters for the Notification Channel

Parameter	Value
ConnectionReliability	BestEffort
StartTimeSupported	false
OrderPolicy	FifoOrder
DiscardPolicy	FifoOrder
MaxEventsPerConsumer[1]	10000

1. The MaxEventsPerConsumer value is configurable. To modify this value, see B.8 Location of the Naming Service IOR File, page B-6. All other parameter values are defined by the TMF and cannot be changed.

CTM sets the RejectNewEvents administration property of the channel to true, as defined by the TMF.

Every notification that CTM generates has the QoS parameters listed in the following table. These parameter values are defined by the TMF and cannot be changed.

Table A-2 QoS Parameters for Notifications

Parameter	Value
EventReliability	BestEffort
Priority	DefaultPriority
Timeout	30 minutes for alarms and TCAs; 24 hours for all other notifications

The NMS retrieves all active alarms on the Element Management System (EMS) and on managed element objects. The NMS filters alarms based on probable cause and managed element name. CTM GateWay/CORBA forwards CTM-specific alarms to the NMS.

For NE-related alarms, CTM inserts the original probable cause in the alarm's nativeProbableCause field. The TMF-mapped probable cause is available in the probableCause field.

A.2 Resolving the Notification Service

CTM GateWay/CORBA can integrate with external notification services. When CTM GateWay/CORBA starts, a reference to EventNotifyChannelFactory is resolved. CTM GateWay/CORBA makes the following attempts to resolve the object:

1. Resolve initial references with the default service name of NotificationService to resolve the object. You can change the service name by setting the corbagw.notification.ServiceName property in the corbagw.properties file.
2. Resolve the object through the naming service. The default naming service entry is services.NotifyChannelFactory. You can change the naming service entry by setting the corbagw.notification.NamingContext property in the corbagw.properties file.
3. Resolve the object directly through the Interoperable Object Reference (IOR). You can change the URL of the IOR file by setting the corbagw.notification.FactoryIORFile property in the corbagw.properties file. The default location for the IOR file is file://localhost/CTM_baseDir/cfg/NotificationService.ior. You can specify a URL that points to a remote web server.

A.3 Tips to Improve Notification Throughput

While CTM GateWay/CORBA and the bundled notification server are developed and tuned to provide high throughput, you might want to take additional steps to improve throughput and avoid any loss of notifications. You should implement the StructuredPushConsumer interface defined by the Object Management Group (OMG) and implement the push_structured_event method. The notification server invokes this method on the NMS.

Caution Because all CORBA calls are blocking, the notification server thread blocks until the push_structured_event method completes. If you try to perform a time-consuming task before the method completes, you will block the notification server.

To improve throughput, you should receive the event and forward it to another thread for processing.

For example:

```
void push_structured_event(StructuredEvent notification) {
    try {
        some_other_thread_queue.addNotification(notification);
    } catch (Exception ex) {
        // catch all exception and ignore
        ex.printStackTrace();
    }
}
```

Caution As shown in the example, you must catch all exceptions, or you will stop receiving notifications.

A.4 Supported Events

The following sections list the attributes of each event.

A.4.1 Object Creation

The following table lists the object creation types and attributes.

Table A-3 Object Creation

Attribute Name	Type	Supported by CTM?
notificatId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
objectType	notifications::ObjectType_T	Yes.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	No.
edgePoint	boolean	No.

Object creation notifications are sent to the NMS for the objects in the following table.

Table A-4 Object Creation Notifications

Object Type	Type Sent in remainder_of_body for Object Creation Event
OT_MANAGED_ELEMENT	managedElement::ManagedElement_T
OT_MULTILAYER_SUBNETWORK	multiLayerSubnetwork::MultiLayerSubnetwork_T
OT_TOPOLOGICAL_LINK	topologicalLink::TopologicalLink_T
OT_SUBNETWORK_CONNECTION	subnetworkConnection::SubnetworkConnection_T
OT_EQUIPMENT	equipment::Equipment_T
OT_PROTECTION_GROUP	protection::ProtectionGroup_T

Table A-4 Object Creation Notifications (continued)

Object Type	Type Sent in remainder_of_body for Object Creation Event
OT_L2TOPOLOGY	MultiLayerSubnetwork::L2Topology_T
OT_MLVLAN	MultiLayerSubnetwork::MLVLAN_T
OT_ROLL	MultiLayerSubnetwork::Roll_T
OT_VCAT	SubnetworkConnection::VCAT_T

CTM does not generate object creation events for SubnetworkConnection for the ONS 15540.

CTM cannot give the object name for object creation events for ONS 15530 and ONS 15540 equipment.

For the MGX 8880/8850/8830, CTM generates object creation events for the object types OT_MANAGED_ELEMENT and OT_EQUIPMENT.

Note The number of equipment holders for NEs supported by CTM is constant. To reduce traffic between CTM and the NMS, CTM does not generate object creation events for equipment holders. The number of PTPs on equipment is also constant. Therefore, CTM generates only object creation events for equipment.

A.4.2 Object Deletion

The following table lists the object deletion types and attributes.

Table A-5 Object Deletion

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
objectType	notifications::ObjectType_T	Yes.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	No.
edgePoint	boolean	No.

Object deletion notifications are sent to the NMS for the objects in the following table.

Table A-6 Object Deletion Notifications

Object Type	Type Sent in remainder_of_body for Object Deletion Event
OT_MANAGED_ELEMENT	managedElement::ManagedElement_T
OT_MULTILAYER_SUBNETWORK	multiLayerSubnetwork::MultiLayerSubnetwork_T
OT_TOPOLOGICAL_LINK	topologicalLink::TopologicalLink_T
OT_SUBNETWORK_CONNECTION	subnetworkConnection::SubnetworkConnection_T
OT_EQUIPMENT	equipment::Equipment_T

Table A-6 Object Deletion Notifications (continued)

Object Type	Type Sent in remainder_of_body for Object Deletion Event
OT_PROTECTION_GROUP	protection::ProtectionGroup_T
OT_L2TOPOLOGY	MultiLayerSubnetwork::L2Topology_T
OT_MLVLAN	MultiLayerSubnetwork::MLVLAN_T
OT_ROLL	MultiLayerSubnetwork::Roll_T
OT_VCAT	SubnetworkConnection::VCAT_T

CTM does not generate object deletion events for SubnetworkConnection for the ONS 15530 and ONS 15540.

For the MGX 8880/8850/8830, CTM generates object deletion events for the object types OT_MANAGED_ELEMENT and OT_EQUIPMENT.

Note The number of equipment holders for all NEs supported by CTM is constant. To reduce traffic between CTM and the NMS, CTM does not generate object deletion events for equipment holders. The number of PTPs on equipment is also constant. Therefore, CTM generates only object deletion events for equipment.

A.4.3 Attribute Value Change

The following table lists the attribute value change (AVC) types and attributes.

Table A-7 Attribute Value Change

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
objectType	notifications::ObjectType_T	Yes.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	No.
edgePoint	boolean	No.
attributeList	notifications::NVList_T	Yes.

Attribute value change notifications are sent to the NMS for the objects and attributes in the following table.

Table A-8 Attribute Value Change Notifications

Object Type	Attributes
OT_EMS	nativeEMSName
OT_MANAGED_ELEMENT	nativeEMSName—This field reflects the new managed element name if it is changed. location version productName
OT_MULTILAYER_SUBNETWORK	NativeEMSName
OT_TOPOLOGICAL_LINK[1]	None
OT_SUBNETWORK_CONNECTION[2]	None
OT_L2TOPOLOGY	NativeEMSName
OT_MLVLAN	None **Note** This event notifies the NMS that the VLAN was updated.
OT_ROLL	None **Note** This event notifies the NMS that the roll was updated.
OT_VCAT	None **Note** This event notifies the NMS that the VCAT changed.

1. The OT_TOPOLOGICAL_LINK event notifies the NMS that some attribute of the topological link changed.
2. The OT_SUBNETWORK_CONNECTION event is generated only when CTM fails to delete an SNC and it remains in the same state.

Note
- CTM does not generate AVC events for SubnetworkConnection for the ONS 15530 and ONS 15540.
- For the MGX 8880/8850/8830, CTM generates AVC events for the object types OT_MANAGED_ELEMENT and OT_EMS.

A.4.4 State Change Event

The following table lists the state change event types and attributes.

Table A-9 State Change Event

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
objectType	notifications::ObjectType_T	Yes.

Table A-9 State Change Event (continued)

Attribute Name	Type	Supported by CTM?
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	No.
edgePoint	boolean	No.
attributeList	notifications::NVList_T	Yes.

State change event notifications are sent to the NMS for the objects and attributes in the following table.

Table A-10 State Change Event Notifications

Object Type	Attributes
OT_MANAGED_ELEMENT	communicationState—CTM supports the following values: • CS_AVAILABLE • CS_UNAVAILABLE emsInSyncState—CTM supports the following values: • True • False
OT_EQUIPMENT_HOLDER	holderState—State change. CTM supports the following values: • EMPTY • INSTALLED_AND_EXPECTED • EXPECTED_AND_NOT_INSTALLED • MISMATCH_OF_INSTALLED_AND_EXPECTED
OT_SUBNETWORK_CONNECTION[1]	sncState—CTM supports the following values: • SNCS_DELETING • SNCS_PARTIAL • SNCS_ACTIVE
OT_L2TOPOLOGY	L2TopoState—CTM supports the following values: • L2TS_COMPLETE • L2TS_INCOMPLETE topoInSyncState—CTM supports the following values: • true • false
OT_MLVLAN	vlanState—CTM supports the following values: • VLS_COMPLETE • VLS_INCOMPLETE

Table A-10 State Change Event Notifications (continued)

Object Type	Attributes
OT_ROLL	rollState—CTM supports the following values: • ROLL_PENDING • ROLL_COMPLETED • ROLL_CANCELLED • ROLL_INCOMPLETE
OT_VCAT	vcatState—CTM supports the following values: • VCATS_DELETING • VCATS_PARTIAL • VCAT_ACTIVE

1. The OT_SUBNETWORK_CONNECTION event is generated only when CTM changes the SNC. If CTC or any other tool is used to change the SNC, CTM does not report this event.

CTM does not generate state change events for EquipmentHolder and SubnetworkConnection for the ONS 15530 and ONS 15540.

For the MGX 8880/8850/8830, CTM generates state change events for the object types OT_MANAGED_ELEMENT and OT_EQUIPMENT_HOLDER.

A.4.5 Protection Switch

The following table lists the protection switch types and attributes. The protection switch event is not supported for MGX 8880/8850/8830 NEs.

Table A-11 Protection Switch

Attribute Name	Type	Supported by CTM?
notificationId	String	Yes. The uniqueness and the sequence of the notification ID are not guaranteed.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	No.
ProtectionType	protection::ProtectionType_T	No.
switchReason	protection::switchReason_T	No.
layerRate	transmissionParameters::LayerRate_T	No.
groupName	globaldefs::NamingAttributes_T	No. (Yes for ONS 15540 and ONS 15530.)
protectedTP	globaldefs::NamingAttributes_T	No. (Yes for ONS 15540 and ONS 15530.)
switchAwayFromTP	globaldefs::NamingAttributes_T	Yes for ONS 15540 and ONS 15530.
switchToTP	globaldefs::NamingAttributes_T	No. (Yes for ONS 15540 and ONS 15530.)

Table A-11 Protection Switch (continued)

Attribute Name	Type	Supported by CTM?
affectedTP **Note** This is not defined by the TMF.	globaldefs::NamingAttributes_T	Yes. (No for ONS 15540 and ONS 15530.)
Description **Note** This is not defined by the TMF.	String	Yes. The value of this field is the native NE event value.

A.4.6 Threshold Crossing Alert

The following table lists the TCA types and attributes. This event is not supported for ONS 15530, ONS 15540, and MGX 8880/8850/8830 NEs because TCAs are reported as alarms.

Table A-12 Threshold Crossing Alert

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and the sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
nativeEMSName	string	Yes.
objectType	notifications::ObjectType_T	Yes.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	Yes.
isClearable	boolean	Yes. Always set to false.
perceivedSeverity	notifications::PerceivedSeverity_T	Yes. Always set to PerceivedSeverity_T.PS_INDETERMINATE.
layerRate	transmissionParameters::LayerRate_T	Yes.
granularity	Granularity_T	Yes.
pmParameterName	PMParameterName_T	Yes. **Note** If the NE does not report the location, this field is empty.
pmLocation	PMLocation_T	Yes. **Note** If the NE does not report the location, this field is empty.
thresholdType	PMThresholdType_T	No.
value	float	Yes.
unit	string	No.

A.4.7 Alarm

The following table lists the most common alarm messages.

If there is an alarm on an STS/VC4 CTP utilizing all ports of a DS1/E1 card, the alarm is reported with the containing PTP as port 127. If an STS/VC4 SNC exists with one of the drop CTPs of the SNC using all ports on a DS1/E1 card, and if there is an alarm on the STS CTP, CTM reports the port number of the containing PTP as 127.

Table A-13 Alarm

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
objectName	globaldefs::NamingAttributes_T	Yes.
nativeEMSName	string	Yes.
nativeProbableCause	string	Yes. **Note** In case of external condition alarms, the string value of the condition type is appended to this field.
objectType	notifications::ObjectType_T	Yes.
emsTime	globaldefs::Time_T	Yes.
neTime	globaldefs::Time_T	Yes.
isClearable	boolean	Yes. The value is always set to True.
layerRate	transmissionParameters::LayerRate_T	Yes. **Note** Not supported for ONS 15530 and ONS 15540 NEs.
probableCause	string	Yes.
probableCauseQualifier	string	Yes. CTM provides the same value as in the nativeProbableCause field.
perceivedSeverity	notifications::PerceivedSeverity_T	Yes.
serviceAffecting	notifications::ServiceAffecting_T	Yes.
affectedTPList	globaldefs::NamingAttributesList_T	Yes.
additionalText	string	Yes.
additionalInfo	globaldefs::NVSList_T	Yes. CTM provides subnetwork name when available. If the alarmed object is OC-n PTP, CTM also includes SDH_SONET_SS_BITS in this list. CTM provides the IP address for ME-related alarms. CTM provides the ME name when the object type is OT_EMS and the alarm is ME-related. CTM provides the affected object when the object type is OT_EMS and the nativeProbableCause is “Server Monitor Threshold Crossed.”

Table A-13 Alarm (continued)

Attribute Name	Type	Supported by CTM?
isPropagated	boolean	Yes. For the MGX 8880/8850/8830, values are assigned to this field. For all other NEs, this field is always set to False.
aresFDN	string	Yes.

A.4.8 Heartbeat Event

CTM generates a periodic heartbeat event and pushes the event to the notification channel. By default, this function is disabled. You can configure the interval value through the CTM Control Panel. If the interval value is set to 0, the heartbeat event is disabled. By monitoring this event, the northbound interface client knows whether the notification service is active. The type_name field contains the NT_HEART_BEAT value under the fixed_header for this type of event. Filterable_data has only one field, which is shown in the following table.

Table A-14 Heartbeat Event

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.

A.4.9 Backup Status Event

The backup status event contains details about changes to a managed element's backup status.

Table A-15 Backup Status Event

Attribute Name	Type	Supported by CTM?
notificationId	string	Yes. The uniqueness and sequence of the notification ID are not guaranteed.
emsTime	globaldefs::Time_T	Yes. The time when the EMS reported the event.
neTime	globaldefs::Time_T	Yes. The time provided by the NE. If the NE does not report time, this field is optional or reports an empty string.
backupStatus	softwareManager::BackupStatus_T	Yes. The backup status of the managed element.
meName	globaldefs::NamingAttributes_T	Yes. The name of the managed element from which the backup was taken.

A.5 Probable Cause Mapping

A.5.1 TMF-Defined Probable Cause

The following report lists the probable causes defined by the TMF:

"UNIDENTIFIED": for alarms that do not match any other string below. EMS shall in this case fill out the additional text field as much as possible.
"AIS": alarm indication signal
"AMS": Alternate modulation signal
"AU-AIS": AU alarm indication signal
"BER_SD": signal degrade (includes receiver degrade)
"BER_SF": signal fail (includes receiver failure and excessive BER)
"DCC_FAILURE": Data Communication Channel Failure
"EMS": EMS system alarm
"EMS_ALM_LOSS": The 1st notification that the EMS may supply after 1 or more notifications for protection switch, TCA, alarm, or file transfer status have been discarded by the EMS while other events have not been discarded
"EMS_LIFECYCLE_LOSS": The 1st notification that an EMS may supply after 1 or more events of type OC/OD/AVC/SC/RC have been discarded by the EMS
"EMS_ALM_AND_LIFECYCLE_LOSS": The 1st notification that an EMS may supply after 1 or more notifications for protection switch, TCA, alarm, or file transfer status, and 1 or more events of type OC/OD/AVC/SC/RC have been discarded by the EMS
"EQPT": equipment alarm
"ENV" Environmental/external cause
"FOP_APS": failure of APS protocol
"LCD": Loss of Cell Delineation (from TC Adaptor part of ATM NI)
"LOF": loss of frame (when distinguished from LOS)
"LOM": loss of multiframe (SDH only, since not an alarm in GR-253)
"LOP": loss of pointer
"LOS": loss of signal
"MS-AIS": MS alarm indication signal
"OSC-AIS": Optical Supervisory Channel alarm indication signal
"OSC_BER_SF": Optical Supervisory Channel signal fail/RX fail/excessive BER
"OSC_FERF": Optical Supervisory Channel Far End Receive Failure
"OSC_LOF": Optical Supervisory Channel Loss of Frame
"OSC_LOS": Optical Supervisory Channel Loss of Signal
"OSC_SD": Optical Supervisory Channel signal degrade
"PLM": payload label mismatch (when reported as an alarm)
"RAI": remote alarm indication (sometimes reported as RDI)
"SECURITY_VIOLATION": security violation
"SSF": server signal fail
"TCM-AIS": Tandem Connection Sink - Incoming Alarm Indication Signal
"TCM-LOS": Tandem Connection Sink - Loss of Tandem Connection Signal
"TCM-OAI": Tandem Connection Sink - Outgoing Defect Indication (same/similar to Alarm Indication)
"TCM-RAI": Tandem Connection Sink - Remote Defect Indication (same/similar to Alarm Indication)
"TCM-SD": Tandem Connection Sink - Signal Degrade
"TCM-SSF": Tandem Connection Sink - Server Signal Fail
"TCM-TIM": Connection Sink - Trace Identifier Mismatch
"TCM-UNEQ": Tandem Connection Sink - Unequipped

```
"TIM": trace identifier mismatch (when reported as an alarm)
"TU-AIS": TU alarm indication signal
"TX_DEGRADE": transmitter degrade, including laser degrade
"TX_FAIL": transmitter failure, including laser failure
"UAT": Unavailable Time
"UNEQ": payload unequipped
"VC-AIS": VCL/VCC TP Alarm Indication Signal
"VC-RDI": VCL/VCC TP Remote Defect Indication
"VP-AIS": VPL/VPC TP Alarm Indication Signal
"VP-RDI": VPL/VPC TP Remote Defect Indication
```

A.5.2 CTM GateWay/CORBA Mapping—EMS

The following table lists the CTM GateWay/CORBA mapping for the EMS.

Table A-16 CTM GateWay/CORBA Mapping for the EMS

TMF Mapping	NE Native Probable Cause
EMS	Loss of communication
EMS	Memory automatic or manual backup failure
SECURITY_VIOLATION	Maximum login attempts exceeded
EMS	Partition 0...6 free disk space low
EMS	Alarm resync unsuccessful
EMS	Server monitor threshold crossed

A.5.3 CTM GateWay/CORBA Mapping—MGX 8880/8850/8830

The following table lists the CTM GateWay/CORBA mapping for the MGX 8880/8850/8830.

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830

TMF Mapping	NE Native Probable Cause
EMS	Sync-up has not started yet
EMS	Currently in sync-up
EMS	Partial sync-up
EMS	Sync-up failed
EMS	Server in partial sync-up
EMS	Server sync-up failed
Note The following EMS alarms must be cleared manually, so “isClearable” should be set to “False” in the corresponding CTM GateWay/CORBA alarm event.	
EMS	FTP transfer failure
EMS	FTP file size mismatch
EMS	FTP transfer failed
EMS	Upload file error
EMS	Minus 2 trap

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
EMS	Card lost trap
EMS	SNMP retry exceeded
EMS	FTP retry exceeded
EMS	Stats file error
EMS	Stats file transfer error
EMS	SNMP throttle error
EMS	Backoff failed
EMS	SNMP timeout
EMS	FTP session timeout
EMS	FTP transfer timeout
EMS	Unknown error
EMS	Initialization error
EMS	Communication error in ILOG
EMS	Communication error in shared memory
EMS	Communication error in CORBA
EMS	NTS registration failed
EMS	NTS trap loss
EMS	PM error
EMS	Process restarted
Shelf-Level Peripheral Alarms	
EQPT	Above normal
EQPT	Below normal
EQPT	Down
EQPT	Power failure
Card-Level Alarms for all Supported Cards on MGX8880/8850/8830	
EQPT	Front card not present
EQPT	Front card failed
EQPT	Front card in self test
EQPT	Front card held in reset
EQPT	Front card in boot
EQPT	Front card mismatch
EQPT	Front card unknown alarm
EQPT	Front card core card mismatch
EQPT	Front card failed, no backup
EQPT	Front card reserved
EQPT	Front card down

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Front card in update mode
EQPT	Front card unavailable
EQPT	Front card downloading
EQPT	Front card downloader
EQPT	Front card downloaded
EQPT	Front card locked
EQPT	Front card in program mode
EQPT	Front card upgrading
EQPT	Front card upgraded
EQPT	Front card frozen
EQPT	Front card in hold mode
EQPT	Front card not responding
EQPT	Front card in initialization
EQPT	Front card blocked
EQPT	Front card removed
EQPT	Primary back card not present
EQPT	Primary back card in mismatch
EQPT	Secondary back card not present
EQPT	Secondary back card in mismatch
EQPT	Primary card not present
EQPT	Primary card failed
EQPT	Primary card in self test
EQPT	Primary card held in reset
EQPT	Primary card in boot
EQPT	Primary card mismatch
EQPT	Primary card unknown alarm
EQPT	Primary card core card mismatch
EQPT	Primary card failed, no backup
EQPT	Primary card reserved
EQPT	Primary card down
EQPT	Primary card in update mode
EQPT	Primary card unavailable
EQPT	Primary card downloading
EQPT	Primary card downloader
EQPT	Primary card downloaded
EQPT	Primary card locked

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Primary card in program mode
EQPT	Primary card upgrading
EQPT	Primary card upgraded
EQPT	Primary card frozen
EQPT	Primary card in hold mode
EQPT	Primary card not responding
EQPT	Primary card in initialization
EQPT	Primary card blocked
EQPT	Front card removed
EQPT	Secondary card not present
EQPT	Secondary card failed
EQPT	Secondary card in self test
EQPT	Secondary card held in reset
EQPT	Secondary card in boot
EQPT	Secondary card mismatch
EQPT	Secondary card unknown alarm
EQPT	Secondary card core card mismatch
EQPT	Secondary card failed, no backup
EQPT	Secondary card reserved
EQPT	Secondary card down
EQPT	Secondary card in update mode
EQPT	Secondary card unavailable
EQPT	Secondary card downloading
EQPT	Secondary card downloader
EQPT	Secondary card downloaded
EQPT	Secondary card locked
EQPT	Secondary card in program mode
EQPT	Secondary card upgrading
EQPT	Secondary card upgraded
EQPT	Secondary card frozen
EQPT	Secondary card in hold mode
EQPT	Secondary card not responding
EQPT	Secondary card in initialization
EQPT	Redundant blocked
EQPT	Secondary card removed
EQPT	Controller in standby mode

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
EQPT	License expired
EQPT	License missing
EQPT	Card sync-up failed
EQPT	Redundant card switch over
Sensor Alarms: VXSM	
EQPT	Sensor less than threshold value
EQPT	Sensor greater than threshold value
EQPT	Sensor equal to threshold value
EQPT	Sensor less than threshold value
EQPT	Sensor greater than threshold value
EQPT	Sensor equal to threshold value
EQPT	Sensor less than threshold value
EQPT	Sensor greater than threshold value
EQPT	Sensor equal to threshold value
Line-Level Alarms	
DS1 Line: AXSM, MPSM, PXM1E, VXSM, VISM, SRME	
RAI	Receiving RAI
RAI	Transmitting RAI
AIS	Receiving AIS
AIS	Transmitting AIS
OOF	Receiving OOF
LOS	Receiving LOS
UNIDENTIFIED	Near-end local loop in effect
UNIDENTIFIED	Near-end remote loop in effect
UNIDENTIFIED	Near-end remote payload loop in effect
OOF	Receiving CRC MF OOF
OOF	Receiving signal MF OOF
RAI	Receiving MF RAI
AIS	Receiving MF AIS
UNIDENTIFIED	Far-end remote loopback
UNIDENTIFIED	Detected near-end remote loopback
DS3 Line: AXSM, MPSM, PXM1E, VXSM, SRME	
UNIDENTIFIED	Receiving RAI
RAI	Transmitting RAI
AIS	Receiving AIS
AIS	Transmitting AIS

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
OOF	Receiving OOF
LOS	Receiving LOS
UNIDENTIFIED	Near-end local loop in effect
UNIDENTIFIED	Near-end remote loop in effect
UNIDENTIFIED	Receiving test pattern
UNIDENTIFIED	Receiving LOCD
UNIDENTIFIED	Receiving C-bit parity pattern
SONET Line: AXSM, MPSM, PXM1E, VXSM, SRME	
AIS	Line receiving AIS
UNIDENTIFIED	Line receiving RDI
AIS	Line transmitting AIS
UNIDENTIFIED	Line transmitting RDI
LOS	Section LOS
LOF	Section LOF
UNIDENTIFIED	Section trace ID mismatch
LOP	Path receiving STS LOP
AIS	Path receiving STS AIS
UNIDENTIFIED	Path receiving STS RDI
UNEQ	Path unequipped
UNIDENTIFIED	Path signal label mismatch
UNIDENTIFIED	Path trace failure
AIS	Path transmitting AIS
UNIDENTIFIED	Path transmitting RDI
UNIDENTIFIED	Path signal receiving LOCD
Line Stats Alarm	
DS1/DS3: AXSM, MPSM, PXM1E, VXSM, VISM, SRME	
UNIDENTIFIED	Stats alarm, PES current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, PES 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, PSES current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, PSES 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, SEFS current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, SEFS 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, UAS current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, UAS 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, LCV current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, LCV 24-hour threshold exceeded

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Stats alarm, PCV current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, PCV 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, LES current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, LES 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, CCV current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, CCV 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, CES current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, CES 24-hour threshold exceeded
UNIDENTIFIED	Stats alarm, CSES current 15-minute threshold exceeded
UNIDENTIFIED	Stats alarm, CSES 24-hour threshold exceeded
SONET: AXSM, MPSM, PXM1E, VXSM, SRME	
UNIDENTIFIED	Stats alarm, SONET line, total ESs
UNIDENTIFIED	Stats alarm, SONET line, total SESs
UNIDENTIFIED	Stats alarm, SONET line, total CVs
UNIDENTIFIED	Stats alarm, SONET line, total UAs
UNIDENTIFIED	Stats alarm, SONET line, current ESs
UNIDENTIFIED	Stats alarm, SONET line, current SESs
UNIDENTIFIED	Stats alarm, SONET line, current CVs
UNIDENTIFIED	Stats alarm, SONET line, current UAs
UNIDENTIFIED	Stats alarm, SONET line, far end, current ESs
UNIDENTIFIED	Stats alarm, SONET line, far end, total ESs
UNIDENTIFIED	Stats alarm, SONET line, far end, total SESs
UNIDENTIFIED	Stats alarm, SONET line, far end, total CVs
UNIDENTIFIED	Stats alarm, SONET line, far end, total UAs
UNIDENTIFIED	Stats alarm, SONET line, far end, current SESs
UNIDENTIFIED	Stats alarm, SONET line, far end, current CVs
UNIDENTIFIED	Stats alarm, SONET line, far end, current UASs
UNIDENTIFIED	Stats alarm, SONET section, total ESs
UNIDENTIFIED	Stats alarm, SONET section, total SESs
UNIDENTIFIED	Stats alarm, SONET section, total SEFSs
UNIDENTIFIED	Stats alarm, SONET section, total CVs
UNIDENTIFIED	Stats alarm, SONET section, current ESs
UNIDENTIFIED	Stats alarm, SONET section, current SESs
UNIDENTIFIED	Stats alarm, SONET section, current SEFSs
UNIDENTIFIED	Stats alarm, SONET section, current CVs
UNIDENTIFIED	Stats alarm, SONET path, total ESs

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Stats alarm, SONET path, total SESs
UNIDENTIFIED	Stats alarm, SONET path, total CVs
UNIDENTIFIED	Stats alarm, SONET path, total UAs
UNIDENTIFIED	Stats alarm, SONET path, current ESs
UNIDENTIFIED	Stats alarm, SONET path, current SESs
UNIDENTIFIED	Stats alarm, SONET path, current CVs
UNIDENTIFIED	Stats alarm, SONET path, current UAs
UNIDENTIFIED	Stats alarm, SONET path, far end, current ESs
UNIDENTIFIED	Stats alarm, SONET path, far end, total ESs
UNIDENTIFIED	Stats alarm, SONET path, far end, total SESs
UNIDENTIFIED	Stats alarm, SONET path, far end, total CVs
UNIDENTIFIED	Stats alarm, SONET path, far end, total UAs
UNIDENTIFIED	Stats alarm, SONET path, far end, current SESs
UNIDENTIFIED	Stats alarm, SONET path, far end, current CVs
UNIDENTIFIED	Stats alarm, SONET path, far end, current UASs
Line APS Alarms: AXSM, MPSM, PXM1E, VXSM, SRME	
FOP_APS	APS alarm, channel mismatch
FOP_APS	APS alarm, protection byte fail
FOP_APS	APS alarms, FE protection failure
FOP_APS	APS alarms, mode mismatch
FOP_APS	APS alarms, signal degrade
FOP_APS	APS alarms, signal failure
FOP_APS	APS alarms, line alarm
FOP_APS	APS alarms, line loopback
FOP_APS	APS alarms, standby signal degrade
FOP_APS	APS alarms, standby signal failure
FOP_APS	APS alarms, standby line alarm
FOP_APS	APS alarms, standby line loopback
FOP_APS	APS alarms, direction mismatch
FOP_APS	APS alarms, wrong request
EQPT	APS alarms, protect BC missing
EQPT	APS alarms, protect DC missing
EQPT	APS alarms, working BC missing
EQPT	APS alarms, working DC missing
EQPT	APS alarms, working card mismatch
EQPT	APS alarms, protect card mismatch

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
FOP_APS	APS alarms, APS revertive switch fail
FOP_APS	APS alarms, APS manual switch fail
FOP_APS	APS alarms, APS SD low switch fail
FOP_APS	APS alarms, APS SD high switch fail
FOP_APS	APS alarms, APS SF low switch fail
FOP_APS	APS alarms, APS SF high switch fail
FOP_APS	APS alarms, APS force switch fail
FOP_APS	APS alarms, APS lockout switch fail
FOP_APS	APS alarms, APS line status OK
Port Alarms: AXSM, MPSM, PXM1E	
UNIDENTIFIED	Port, not configured
UNIDENTIFIED	Port, down/failure
UNIDENTIFIED	Port, remote loopback
UNIDENTIFIED	Port, signaling failure
UNIDENTIFIED	Port, out of cell delineation
UNIDENTIFIED	Port, lower layer down
RPM Port Alarms: RPM, RPM-XF	
UNIDENTIFIED	RPM port, down
UNIDENTIFIED	RPM port, testing
UNIDENTIFIED	RPM port, unknown
UNIDENTIFIED	RPM port, lower layer down
SVC Port Alarms: AXSM, VXSM, RPM-XF, RPM	
UNIDENTIFIED	SVC port, failed
UNIDENTIFIED	SVC port, inactive
UNIDENTIFIED	SVC port, lower layer down
VISM Port Alarms: VISM	
UNIDENTIFIED	VISM port, failed
UNIDENTIFIED	VISM port, degraded
UNIDENTIFIED	VISM port, lower layer down
Aggregate Port Alarms: AXSM, MPSM, PXM1E RPM, RPM-XF	
UNIDENTIFIED	Aggregate port alarm; one or more connections on this port are in primary failure
UNIDENTIFIED	Aggregate port alarm; one or more connections on this port are in secondary failure
UNIDENTIFIED	Aggregate port alarm; one or more connections on this port are incomplete
Connection Alarm: AXSM, MPSM, PXM1E, VXSM, VISM, RPM, RPM-XF	

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Connection alarm, primary failure
UNIDENTIFIED	Connection alarm, secondary failure
UNIDENTIFIED	Incomplete connection
Card-Level VXSM and VISM Alarms	
UNIDENTIFIED	CAC resources, above high threshold value
UNIDENTIFIED	CAC resources, above medium threshold value
UNIDENTIFIED	Media gateway link state, pending in service
UNIDENTIFIED	Media gateway link state, pending FOOS
UNIDENTIFIED	Media gateway link state, pending GOOS
UNIDENTIFIED	Media gateway link state, out of service
UNIDENTIFIED	Media gateway link state, MGC direct out of service
UNIDENTIFIED	RUDP session state, out of service
UNIDENTIFIED	RUDP session state, standby in service
UNIDENTIFIED	RUDP session state, full in service
UNIDENTIFIED	RUDP session state, unknown
UNIDENTIFIED	Session group state, out of service
UNIDENTIFIED	Session group state, unknown
UNIDENTIFIED	Session set state, out of service
UNIDENTIFIED	Session set state, standby in service
UNIDENTIFIED	Session set state, full in service
UNIDENTIFIED	Session set state, unknown
UNIDENTIFIED	MGC in redundancy group, inactive
UNIDENTIFIED	MGC communication state, inactive
UNIDENTIFIED	RTP connection state, failed
UNIDENTIFIED	CID state, failed
UNIDENTIFIED	MTP link set active
UNIDENTIFIED	MTP link set inactive
UNIDENTIFIED	MTP link set inhibited
UNIDENTIFIED	MTP link set uninhibited
UNIDENTIFIED	MTP link set shut
UNIDENTIFIED	MTP link set no shut
UNIDENTIFIED	MTP link active
UNIDENTIFIED	MTP link inactive
UNIDENTIFIED	MTP link inhibited
UNIDENTIFIED	MTP link uninhibited
UNIDENTIFIED	MTP link shut

Table A-17 CTM GateWay/CORBA Mapping for the MGX 8880/8850/8830 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	MTP link no shut
UNIDENTIFIED	MTP signaling gateway route available
UNIDENTIFIED	MTP signaling gateway route unavailable
UNIDENTIFIED	MTP signaling gateway route restricted
UNIDENTIFIED	L3UA ASP active
UNIDENTIFIED	L3UA ASP inactive
UNIDENTIFIED	L3UA ASP down
UNIDENTIFIED	L3UA ASP undefined
UNIDENTIFIED	L3UA AS active
UNIDENTIFIED	L3UA AS inactive
UNIDENTIFIED	L3UA AS down
UNIDENTIFIED	L3UA AS pending
UNIDENTIFIED	L3UA AS undefined
UNIDENTIFIED	DPNSS DLC active
UNIDENTIFIED	DPNSS DLC inactive
UNIDENTIFIED	Gateway voice quality alert
UNIDENTIFIED	Gateway VBD call quality alert
UNIDENTIFIED	Gateway voice quality alert for IP-IP
UNIDENTIFIED	Gateway VQM trap queue full
UNIDENTIFIED	System resource utilization threshold exceeded
UNIDENTIFIED	System resource utilization normal
UNIDENTIFIED	System resource utilization medium level exceeded
UNIDENTIFIED	Zero percent call reduce rate reached
UNIDENTIFIED	Maximum call reduce rate reached
UNIDENTIFIED	Gateway drop call starts
UNIDENTIFIED	Gateway drop call alert cleared
UNIDENTIFIED	Gateway overload normal
UNIDENTIFIED	Gateway overload threshold exceeded
Line-Level VISM and VXSM Alarms	
UNIDENTIFIED	Endpoint state, failed
UNIDENTIFIED	Endpoint state, degraded
UNIDENTIFIED	LAPD state, inactive
UNIDENTIFIED	LAPD trunk state, inactive
UNIDENTIFIED	LAPD trunk state, unknown

A.5.4 CTM GateWay/CORBA Mapping—ONS 15216

The following table lists the alarms for the ONS 15216 OADM R2.2.

Table A-18 CTM GateWay/CORBA Alarms for the ONS 15216 OADM R2.2

TMF Mapping	NE Native Probable Cause
LOS	Add channel signal loss
LOS	Drop channel signal loss
EQPT	Add and drop channels are out of tolerance
EQPT	Add channel signal is too weak and VOA cannot adjust it to match the drop signal
EQPT	Power unit A failed
EQPT	Power unit B failed

A.5.4.1 ONS 15216 EDFA2

The following table lists the alarms for the ONS 15216 EDFA2 R2.3 and 2.4.

Table A-19 CTM GateWay/CORBA Alarms for the ONS 15216 EDFA2 R2.3, R2.4

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Unknown alarm
EQPT	Excessive pump 1 current
EQPT	Excessive pump 2 current
EQPT	Excessive pump 1 laser temperature
EQPT	Excessive pump 2 laser temperature
EQPT	Loss of output power
EQPT	Loss of input power
EQPT	Gain out of range
EQPT	Case temperature out of range
EQPT	Power unit A failed
EQPT	Power unit B failed
EQPT	Database backup
EQPT	Database restore
EQPT	Cutover reset
EQPT	Software load
EQPT	Software reset

A.5.4.2 ONS 15216 EDFA3

The following table lists the alarms for the ONS 15216 EDFA3 R1.0.

Table A-20 CTM GateWay/CORBA Alarms for the ONS 15216 EDFA3 R1.1

TMF Mapping	NE Native Probable Cause	CTM Internal Index (Not Reported in the Notification)
UNIDENTIFIED	Unknown alarm	8701
EQPT	Gain degrade high	8702
EQPT	Gain degrade low	8703
EQPT	Laser 1 bias degrade	8704
EQPT	Laser 2 bias degrade	8705
EQPT	Laser 1 bias fail	8706
EQPT	Laser 2 bias fail	8707
EQPT	Power degrade high LINE1TX port	8708
EQPT	Power degrade low LINE1TX port	8709
EQPT	Power fail low LINE1RX port	8710
EQPT	Power fail low LINE1TX port	8711
EQPT	Power fail low LINE2RX port	8712
EQPT	VOA degrade high	8713
EQPT	VOA degrade low	8714
EQPT	VOA fail high	8715
EQPT	Case temperature out of range	8716
EQPT	Fiber temperature out of range	8717
EQPT	Excessive pump 1 temperature	8718
EQPT	Excessive pump 2 temperature	8719
EQPT	Backup/restore in progress	8720
EQPT	Data integrity fault	8721
EQPT	Equipment failure	8722
EQPT	Exceeding memory capacity	8723
EQPT	Exceeding flash file system capacity	8724
EQPT	Module communication failure	8725
EQPT	Power bus A alarm	8726
EQPT	Power bus B alarm	8727

A.5.5 CTM GateWay/CORBA Mapping—ONS 15305 and ONS 15305 CTC

The following table lists the CTM GateWay/CORBA mapping for the ONS 15305 and ONS 15305 CTC.

Table A-21 CTM GateWay/CORBA Mapping for the ONS 15305, ONS 15305 CTC

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	ONS 15305 condition unknown
AIS	Alarm indication signal
AIS	Alarm indication signal network side
LOF	Alarm indication signal customer side
LOF	Loss of frame alignment
LOF	Loss of frame alignment network side
LOF	Loss of frame alignment customer side
LOP	Loss of pointer
LOS	Loss of signal
BER_SD	Degraded signal defect
LOM	Loss of multiframe alignment
RAI	Remote defect indication
TIM	Trace identifier mismatch
PLM	Payload mismatch
UNEQ	Unequipped
BER_SF	Excessive error defect
UNIDENTIFIED	Communication subsystem failure
SSF	Server signal failure
UNIDENTIFIED	Alarm condition detected on an alarm input port
EQPT	Device main unit failure
EQPT	High temperature alarm
EQPT	Fan failure
EQPT	Power input failure
EQPT	Power output failure
EQPT	Module mismatch
EQPT	Module removed
EQPT	Module failure
UNIDENTIFIED	MSP signaling problem
UNIDENTIFIED	Power module failure
UNIDENTIFIED	LAN port is not working
UNIDENTIFIED	WAN port is not working
UNIDENTIFIED	Too large delay on WAN channel
UNIDENTIFIED	Sequence number fail on WAN channel

Table A-21 CTM GateWay/CORBA Mapping for the ONS 15305, ONS 15305 CTC (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	No synchronization source available
UNIDENTIFIED	Defecting hardware impacting internal T0 clock
UNIDENTIFIED	No T4 synchronization source available with QL >= QL min
UNIDENTIFIED	Inventory failure
UNIDENTIFIED	Diagnostic failure
UNIDENTIFIED	DXC inlet failure
UNIDENTIFIED	DXC inlet bit error
UNIDENTIFIED	Card isolated
UNIDENTIFIED	Card anomaly
UNIDENTIFIED	Hot swap failure
UNIDENTIFIED	Power input failure
UNIDENTIFIED	Transmit degrade on laser
UNIDENTIFIED	Transmit fail on laser
UNIDENTIFIED	Port activated without mapping to an available VC-12
UNIDENTIFIED	Differential VC12 delay for the WAN port is greater than +/- 2 ms
UNIDENTIFIED	Rx buffer overflow in LAN or link interface
UNIDENTIFIED	Interport queue overflow in LAN or link interface
UNIDENTIFIED	Reset of the router/bridge required
UNIDENTIFIED	Device aborted FTP session
UNIDENTIFIED	Automatic switchover to backup link, main link fault
UNIDENTIFIED	Overflow layer 2 forward table
UNIDENTIFIED	Connection failed: frame relay switch and WAN gate
UNIDENTIFIED	Connection established: frame relay switch and WAN gate
UNIDENTIFIED	Error occurred during initialization
UNIDENTIFIED	Overflow in the client table
UNIDENTIFIED	Server not responding to dispatcher polling
UNIDENTIFIED	SNMP SET request was rejected
UNIDENTIFIED	Ping sequence completed
UNIDENTIFIED	Backup taken over for main server or server up after failure
UNIDENTIFIED	DHCP failed to allocate IP address to requesting host
UNIDENTIFIED	Overflow IGMP table
UNIDENTIFIED	Overflow PIM table
UNIDENTIFIED	Overflow condition in routing table
UNIDENTIFIED	Open gate IPX RIP table overflow
UNIDENTIFIED	Open gate IPX SAP table overflow
UNIDENTIFIED	FACS state NE with operation blockAndReport

Table A-21 CTM GateWay/CORBA Mapping for the ONS 15305, ONS 15305 CTC (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Zero hop routing connections table overflow
UNIDENTIFIED	No available IP virtual address
UNIDENTIFIED	Virtual IP address appeared as a source IP
UNIDENTIFIED	Source IP address sent an ARP specifying a virtual IP
UNIDENTIFIED	PPP link got an unrecognized secret
UNIDENTIFIED	FR DLCI status change
UNIDENTIFIED	CHAP failed communication
UNIDENTIFIED	IP SFFT overflow
UNIDENTIFIED	IP NFFT overflow
UNIDENTIFIED	IPX SFFT overflow
UNIDENTIFIED	IPX NFFT overflow
UNIDENTIFIED	IPM FFT overflow
UNIDENTIFIED	PAP failed communication
UNIDENTIFIED	Automatic switchover to backup link, main link fault
UNIDENTIFIED	Informational event
UNIDENTIFIED	T0 sync switch
UNIDENTIFIED	T0QI failed
UNIDENTIFIED	T0QIDnu
UNIDENTIFIED	Switch to protection
UNIDENTIFIED	Switch to working
UNIDENTIFIED	MSP command timeout
UNIDENTIFIED	MSP command overruled
UNIDENTIFIED	Loop closed
UNIDENTIFIED	Loop opened
UNIDENTIFIED	Link up
UNIDENTIFIED	Link down
UNIDENTIFIED	The communication link has come up
UNIDENTIFIED	Failure in communication link
UNIDENTIFIED	Addressee of a protocol message not properly authenticated
UNIDENTIFIED	Reinitializing; configuration or the protocol entity implementation may be altered
UNIDENTIFIED	Reinitializing; neither configuration nor protocol entity implementation altered
UNIDENTIFIED	RMON alarm has crossed the rising threshold
UNIDENTIFIED	RMON alarm has crossed the falling threshold
UNIDENTIFIED	Loss of an adjacency with a PIM neighbor
UNIDENTIFIED	Device finished TFTP transaction

Table A-21 CTM GateWay/CORBA Mapping for the ONS 15305, ONS 15305 CTC (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Device initiated TFTP transaction
UNIDENTIFIED	Auto configuration completed successfully
UNIDENTIFIED	VLAN port dynamically added
UNIDENTIFIED	VLAN port dynamically changed
UNIDENTIFIED	Physical description device changed
UNIDENTIFIED	Port transition from learning to forwarding
UNIDENTIFIED	Port transition from forwarding to learning
UNIDENTIFIED	Packet drop due to the QoS policy
UNIDENTIFIED	Packet forwarded based on the QoS policy
UNIDENTIFIED	Protection link activated
UNIDENTIFIED	Working link activated
UNIDENTIFIED	MSP command timeout
UNIDENTIFIED	MSP command overruled
UNIDENTIFIED	Loss of external synchronization
UNIDENTIFIED	ONS 15305 condition unknown

A.5.6 CTM GateWay/CORBA Mapping—ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH

The following table lists the CTM GateWay/CORBA mapping for the ONS 15454 SONET R4.1.4.

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Normal condition
ENV	Failure detected external to the NE
ENV	External error
UNIDENTIFIED	Excessive switching
UNIDENTIFIED	Incoming failure condition
AIS	Alarm indication signal
AIS	Alarm indication signal—line
AIS	Alarm indication signal—path
FOP_APS	APS channel failure
FOP_APS	Byte failure
FOP_APS	Protection switching channel match failure
FOP_APS	Automatic protection switch mode mismatch
FOP_APS	Far-end protection line failure

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Bipolar violation
UNIDENTIFIED	Carrier loss on the LAN
UNIDENTIFIED	STS concatenation error
UNIDENTIFIED	Excess collisions on the LAN
UNIDENTIFIED	Facility failure
UNIDENTIFIED	Far-end block error
LOF	Loss of frame
LOP	Loss of pointer
LOP	Loss of pointer—path
LOS	Loss of signal
UNIDENTIFIED	Out of frame
UNIDENTIFIED	Path selector failure
RAI	Remote alarm indication
RAI	Remote failure indication
RAI	Remote failure indication—line
RAI	Remote failure indication—path
BER_SD	Signal degrade
UNIDENTIFIED	Severely errored frame
UNIDENTIFIED	Invalid alarm
BER_SF	Signal failure
UNIDENTIFIED	Signal label mismatch failures
UNIDENTIFIED	Payload defect indication
UNIDENTIFIED	Payload defect indication—path
PLM	Payload label mismatch—path
UNEQ	Unequipped—path
TIMING_SYNCH	Loss of synchronization
TIMING_SYNCH	Out of synchronization
TIMING_SYNCH	Primary synchronization reference failure
TIMING_SYNCH	Secondary synchronization reference failure
TIMING_SYNCH	Third synchronization reference failure
TIMING_SYNCH	Fourth synchronization reference failure
TIMING_SYNCH	Fifth synchronization reference failure
TIMING_SYNCH	Sixth synchronization reference failure
UNIDENTIFIED	Outgoing failure condition
RAI	Remote defect indication—line

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
RAI	Remote defect indication—path
TIMING_SYNCH	Free running synchronization mode
TIMING_SYNCH	Holdover synchronization mode
EQPT	Internal fault
UNIDENTIFIED	Internal error
UNIDENTIFIED	Internal message error
EQPT	Mismatch of equipment and attributes
UNIDENTIFIED	Watchdog timer timeout
ENV	Software fault or failure
ENV	Software fault—data integrity fault
UNIDENTIFIED	Program failure
EQPT	Control equipment failure
EQPT	Control processor failure
EQPT	Working memory failure
EQPT	Interconnection equipment failure
EQPT	Time slot interchange equipment failure
EQPT	Equipment failure
EQPT	High temperature
EQPT	Facility termination equipment failure
EQPT	Automatic laser shutdown
UNIDENTIFIED	Failure to release from protection
EQPT	Receiver failure
EQPT	Transmit failure
UNIDENTIFIED	Failure to switch to protection
EQPT	Equipment unit plug-in
ENV	Internal power failure
ENV	Fuse alarm
TIMING_SYNCH	Synchronization unit failure
EQPT	Synchronization switching equipment failure
EQPT	Equipment unit unplugged
EQPT	Manually caused abnormal condition
UNIDENTIFIED	Manual alarm cutoff
UNIDENTIFIED	Procedural error
EQPT	Improper removal
UNIDENTIFIED	Protection unit not available

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Protection switch
UNIDENTIFIED	Recovery or service protection action has been initiated
UNIDENTIFIED	Automatic system reset
UNIDENTIFIED	Cold restart
UNIDENTIFIED	Forced switch back to working
UNIDENTIFIED	Forced switch to protection
UNIDENTIFIED	Initialization initiated
UNIDENTIFIED	Lockout of protection
UNIDENTIFIED	Lockout of working
UNIDENTIFIED	Manual system reset
UNIDENTIFIED	Manual switch to internal clock
UNIDENTIFIED	Manual switch to primary reference
UNIDENTIFIED	Manual switch to secondary reference
UNIDENTIFIED	Manual switch to third reference
UNIDENTIFIED	Manual switch to fourth reference
UNIDENTIFIED	Manual switch to fifth reference
UNIDENTIFIED	Manual switch to sixth reference
UNIDENTIFIED	Manual switch back to working
UNIDENTIFIED	Manual switch to protection
UNIDENTIFIED	Power fail restart
EQPT	Software download in progress
TIMING_SYNCH	Switch to internal clock
TIMING_SYNCH	Switch to primary reference
TIMING_SYNCH	Switch to secondary reference
TIMING_SYNCH	Switch to third reference
TIMING_SYNCH	Switch to fourth reference
TIMING_SYNCH	Switch to fifth reference
TIMING_SYNCH	Switch to sixth reference
UNIDENTIFIED	Switched back to working
UNIDENTIFIED	Switched to protection
UNIDENTIFIED	Warm restart
UNIDENTIFIED	Ring is in wait-to-restore state
EQPT	Primary nonvolatile backup memory failure
EQPT	Secondary nonvolatile backup memory failure
EQPT	Control bus failure

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Control communications equipment failure
UNIDENTIFIED	Loopback
UNIDENTIFIED	Loopback, network
UNIDENTIFIED	Loopback, terminal
ENV	Fan failure
DCC_FAILURE	SDCC termination failure
UNIDENTIFIED	Loopback facility
EQPT	Payload bus failure to I/O slot 1 to 17 (XCON slot 8)
EQPT	Control bus 1 to 2 failure
EQPT	Invalid MAC address
EQPT	Board failure
EQPT	Diagnostics failure
EQPT	Medium access control failure
UNIDENTIFIED	Duplicate node ID
UNIDENTIFIED	Failure to switch to protection—ring
UNIDENTIFIED	Failure to switch to protection—span
UNIDENTIFIED	Manual switch back to working—ring
UNIDENTIFIED	Manual switch back to working—span
UNIDENTIFIED	Manual switch to protection—ring
UNIDENTIFIED	Manual switch to protection—span
UNIDENTIFIED	Forced switch back to working—ring
UNIDENTIFIED	Forced switch back to working—span
UNIDENTIFIED	Forced switch to protection—ring
UNIDENTIFIED	Forced switch to protection—span
UNIDENTIFIED	Lockout of protection—ring
UNIDENTIFIED	Lockout of protection—span
UNIDENTIFIED	Lockout of working—ring
UNIDENTIFIED	Lockout of working—span
UNIDENTIFIED	Ring is squelching traffic
FOP_APS	Inconsistent APS code
FOP_APS	Node ID mismatch
FOP_APS	Default K byte
FOP_APS	Connection loss
EQPT	TCC [A...B] to shelf slot 1...17 [DROP 1...8 | TRUNK 1...2 | TCC A...B | XCON A...B] communication failure
TIMING_SYNCH	Fast start synchronization mode

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
FOP_APS	Improper APS code
UNIDENTIFIED	BLSR out of sync
AIS	Alarm indication signal—VT
RAI	Remote failure indication—VT
PLM	Signal label mismatch failure—payload label mismatch—VT
UNEQ	Signal label mismatch failure—unequipped—VT
UNIDENTIFIED	Peer state mismatch
EQPT	Facility termination equipment—receiver missing
EQPT	Facility termination equipment—transmitter missing
LOP	Loss of pointer—VT
LINK DOWN	Embedded operations channel failure—link down
TIMING_SYNCH	Failed to receive synchronization status message
TIMING_SYNCH	Synchronization status messages are disabled on this interface
TIMING_SYNCH	Primary reference source—stratum 1 traceable
TIMING_SYNCH	Synchronized—traceability unknown
TIMING_SYNCH	Stratum 2 traceable
TIMING_SYNCH	Transit node clock traceable
TIMING_SYNCH	Stratum 3E traceable
TIMING_SYNCH	Stratum 3 traceable
TIMING_SYNCH	SONET minimum clock traceable
TIMING_SYNCH	Stratum 4 traceable
TIMING_SYNCH	Do not use for synchronization
TIMING_SYNCH	Reserved for network synchronization use
UNIDENTIFIED	Failure to switch to protection—path
UNIDENTIFIED	Manual switch of working facility—equipment to protection—path
UNIDENTIFIED	Working facility—equipment forced to switch to protection—path
UNIDENTIFIED	Lockout of protection—path
UNIDENTIFIED	Automatic UPSR switch caused by [AIS | LOP | UNEQ | PDI | SFBER | SDBER]
UNIDENTIFIED	DS3 idle condition
UNIDENTIFIED	DS1 loopback due to FEAC command
UNIDENTIFIED	DS1 loopback command sent to far end
UNIDENTIFIED	DS3 loopback due to FEAC command
UNIDENTIFIED	DS3 loopback command sent to far end

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	DS2 loopback due to far-end command
UNIDENTIFIED	DS2 loopback command sent to far end
ENV	Far-end AIS
LOS	Far-end multiple DS1 LOS detected on DS3
ENV	Far-end DS1 equipment failure—non-service affecting
ENV	Far-end DS1 equipment failure—service affecting
LOS	Far-end single DS1 LOS
ENV	Far-end DS3 equipment failure—non-service affecting
ENV	Far-end DS3 equipment failure—service affecting
ENV	Far-end common equipment failure—non-service affecting
UNIDENTIFIED	Far end idle
LOS	Far end LOS
LOF	Far end LOF
UNIDENTIFIED	Performance monitor threshold crossing alert
UNIDENTIFIED	Ethernet bridge is new root of spanning tree
UNIDENTIFIED	Ethernet bridge topology change
UNIDENTIFIED	BLSR tables resynchronized
ENV	System reboot
UNIDENTIFIED	Manual switch request on facility—equipment
UNIDENTIFIED	Forced switch request on facility or equipment
UNIDENTIFIED	Lockout switch request on facility—equipment
UNIDENTIFIED	RMON histories and alarms reset reboot
UNIDENTIFIED	RMON alarm
LOS	Video interface card loss of video IF signal
LOS	VIC loss of video baseband signal
LOS	VIC loss of audio FM signal
LOS	VIC loss of audio baseband channel [1...4] signal
TIM	STS path trace identifier mismatch
ENV	NE power failure at connector A
ENV	NE power failure at connector B
EQPT	Free memory on card very low
EQPT	Free memory on card near zero
UNIDENTIFIED	Exercise request on ring
UNIDENTIFIED	Exercise request on span
UNIDENTIFIED	Squelching path

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Extra traffic preempted
UNIDENTIFIED	Far-end lockout of working—ring
UNIDENTIFIED	Far-end lockout of working—span
UNIDENTIFIED	Far-end lockout of protection—ring
UNIDENTIFIED	Far-end lockout of protection—all spans
UNIDENTIFIED	Far-end working facility forced to switch to protection—ring
UNIDENTIFIED	Far-end working facility forced to switch to protection—span
UNIDENTIFIED	Far-end manual switch of working facility to protection—ring
UNIDENTIFIED	Far-end manual switch of working facility to protection—span
UNIDENTIFIED	Far-end exercising ring
UNIDENTIFIED	Far-end exercising span
ENV	Far-end BER threshold passed for signal failure—ring
ENV	Far-end BER threshold passed for signal failure—span
ENV	Far-end BER threshold passed for signal degrade—ring
ENV	Far-end BER threshold passed for signal degrade—span
ENV	APS channel—far-end protection line signal degrade
UNIDENTIFIED	Ring switch is active on the east side
UNIDENTIFIED	Ring switch is active on the west side
UNIDENTIFIED	Span switch is active on the east side
UNIDENTIFIED	Span switch is active on the west side
UNIDENTIFIED	Unidirectional full pass-through is active
UNIDENTIFIED	Bidirectional full pass-through is active
UNIDENTIFIED	K bytes pass-through is active
UNIDENTIFIED	Ring is segmented
UNIDENTIFIED	Ring topology is under construction
UNIDENTIFIED	Lockout of protection—all spans
UNIDENTIFIED	Far-end of fiber is provisioned with different ring ID
UNIDENTIFIED	Both ends of fiber provisioned as east or both as west
SECURITY_VIOLATION	Security—invalid login—see audit trail
UNIDENTIFIED	Autonomous messages inhibited
UNIDENTIFIED	Traffic storm on LAN; LAN temporarily disabled
UNIDENTIFIED	REPT^DBCHG messages inhibited
SECURITY_VIOLATION	Security—user ID has expired
EQPT	Partial fan failure
UNIDENTIFIED	Force switch request on ring

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Force switch request on span
UNIDENTIFIED	Lockout switch request on ring
UNIDENTIFIED	Lockout switch request on span
UNIDENTIFIED	Manual switch request on ring
UNIDENTIFIED	Manual switch request on span
EQPT	Peer-to-peer slot communication failure
EQPT	Peer-to-peer slot communication failure
EQPT	TCC A to shelf slot communication failure
EQPT	TCC B to shelf slot communication failure
EQPT	Interconnection equipment failure—working XC payload bus
EQPT	Interconnection equipment failure—protect XC payload bus
EQPT	Inhibit switch to protect request on equipment
EQPT	Inhibit switch to working request on equipment
BER_SD	BER threshold exceeded for signal degrade—line
BER_SD	BER threshold exceeded for signal degrade—path
BER_SF	BER threshold exceeded for signal failure—line
BER_SF	BER threshold exceeded for signal failure—path
UNIDENTIFIED	Exercising ring successfully
UNIDENTIFIED	Exercising span successfully
UNIDENTIFIED	Span is in wait-to-restore state
EQPT	Peer card not responding
UNIDENTIFIED	Exercise request on ring failed
UNIDENTIFIED	Exercise request on span failed
UNIDENTIFIED	Far-end lockout of protection—span
EQPT	Manufacturing data memory (EEPROM) failure
EQPT	Replaceable equipment/unit is missing
ENV	Software download failure
TIM	TIM section—trace identifier mismatch failure
AIS	AIS—multiplex section—alarm indication signal
RAI	RFI—multiplex section—remote failure/alarm indication
TIM	TIM—high order—trace identifier mismatch failure
AU_AIS	AIS—administration unit—alarm indication signal
LOP	LOP—administration unit—loss of pointer
UNEQ	SLMF—unequipped high order—path unequipped
PLM	SLMF—PLM high order—path label mismatch

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
RAI	RFI—high order—remote failure/alarm indication
LOP	LOP—tributary unit—loss of pointer
TU-AIS	AIS—tributary unit—alarm indication signal
UNEQ	SLMF—unequipped low order—path unequipped
PLM	SLMF—PLM low order—path label mismatch
TIM	TIM low order—trace identifier mismatch failure
RAI	RFI—low order—remote failure/alarm indication
TIMING_SYNCH	G811—primary reference clock traceable
TIMING_SYNCH	G812—transit node clock traceable
TIMING_SYNCH	G812—local node clock traceable
TIMING_SYNCH	G813—synchronous equipment timing source traceable
UNIDENTIFIED	E1 loopback due to FEAC command
UNIDENTIFIED	E1 loopback command sent to far end
UNIDENTIFIED	E3 loopback due to FEAC command
ENV	Far-end multiple E1 LOS detected on E3
ENV	Far-end E1 equipment failure non-service affecting
ENV	Far-end E1 equipment failure service affecting
ENV	Far-end single E1 LOS
ENV	Far-end E3 equipment failure service affecting
UNIDENTIFIED	E3 loopback command sent to far end
ENV	Far-end E3 equipment failure non-service affecting
ENV	Low voltage—battery A
ENV	High voltage—battery A
ENV	Low voltage—battery B
ENV	High voltage—battery B
UNIDENTIFIED	Procedural error—MS-SPRing out of sync
UNIDENTIFIED	MS-SPRing tables resynchronized
UNIDENTIFIED	Automatic SNCP switch caused by [AIS | LOP | UNEQ | PDI | SFBER | SDBER]
UNIDENTIFIED	STM concatenation error
UNIDENTIFIED	E3 idle condition
UNIDENTIFIED	IOS config copy failed
UNIDENTIFIED	IOS config copy in progress
UNIDENTIFIED	Alarms suppressed by user command
UNIDENTIFIED	Alarms suppressed for maintenance
UNIDENTIFIED	N/A

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Synchronization reference frequency out of bounds
UNIDENTIFIED	Ether Tx excess flow control
UNIDENTIFIED	Ether Tx oversubscribed
UNIDENTIFIED	Transport layer failure
UNIDENTIFIED	Ether Rx excess flow control
UNIDENTIFIED	Ether Rx oversubscribed
UNIDENTIFIED	Ether Tx underrun
UNIDENTIFIED	SNTP host failure
UNIDENTIFIED	DS3 frame format mismatch
UNIDENTIFIED	Alarms/events suppressed for this object
BER_SD	BER threshold exceeded for signal degrade—high order
BER_SF	BER threshold exceeded for signal failure—high order
BER_SD	BER threshold exceeded for signal degrade—low order
BER_SF	BER threshold exceeded for signal failure—low order
UNIDENTIFIED	Failure to switch to protection—high-order path
UNIDENTIFIED	Failure to switch to protection—low-order path
UNIDENTIFIED	Wait to restore
ENV	Extreme high voltage—battery A
ENV	Extreme low voltage—battery A
ENV	Extreme high voltage—battery B
ENV	Extreme low voltage—battery B
UNIDENTIFIED	Switching matrix module failure
UNIDENTIFIED	Signaling unable to set up circuit
UNIDENTIFIED	RSVP hello FSM to neighbor down
UNIDENTIFIED	LMP hello FSM to control channel down
UNIDENTIFIED	LMP neighbor discovery has failed
UNIDENTIFIED	Unauthorized incoming signaling request
UNIDENTIFIED	Signaled circuit going down
UNIDENTIFIED	Autonomous PM report message inhibited
EQPT	I/O slot to XCON communication failure
UNIDENTIFIED	Forced switch to primary reference
UNIDENTIFIED	Forced switch to secondary reference
UNIDENTIFIED	Forced switch to third reference
UNIDENTIFIED	Forced switch to internal clock
LOM	LOF—administration unit—loss of multiframe

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Admin logout of user
UNIDENTIFIED	User locked out
UNIDENTIFIED	Admin lockout of user
UNIDENTIFIED	Admin lockout clear
SECURITY_VIOLATION	Security—invalid login—username—see audit log
SECURITY_VIOLATION	Security—invalid login—password—see audit log
SECURITY_VIOLATION	Security—invalid login—locked out—see audit log
SECURITY_VIOLATION	Security—invalid login—already logged on—see audit log
UNIDENTIFIED	Login of user
UNIDENTIFIED	Automatic logout of idle user
UNIDENTIFIED	Logout of user
UNIDENTIFIED	Cross-connect loopback
UNIDENTIFIED	Error in startup config
UNIDENTIFIED	No startup config
EQPT	Laser approaching end of life
UNIDENTIFIED	Database backup failed
UNIDENTIFIED	Database restore failed
UNIDENTIFIED	8B10B out of sync
AIS	ODUk—alarm indication signal
AIS	OTUk—alarm indication signal
UNIDENTIFIED	OTUk—backward defect indicator
UNIDENTIFIED	ODUk—backward defect indicator
UNIDENTIFIED	FEC uncorrected word
UNIDENTIFIED	GCC embedded operation channel failure
UNIDENTIFIED	OTUk—incoming alignment error
UNIDENTIFIED	ODUk—locked defect—PM
LOM	Loss of multiframe
UNIDENTIFIED	ODUk—open connection indication
PLM	Payload type identifier mismatch
TIM	ODUk—trail trace identifier mismatch
TIM	OTUk—trail trace identifier mismatch
EQPT	Equipment high laser bias
EQPT	Equipment high laser temp
EQPT	Equipment high laser Peltier
EQPT	Equipment high Rx power

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Equipment high Tx power
EQPT	Equipment high Rx temperature
EQPT	Equipment low Tx temperature
EQPT	Equipment high transceiver voltage
EQPT	Equipment low laser bias
EQPT	Equipment low laser temp
EQPT	Equipment low laser Peltier
EQPT	Equipment low Rx power
EQPT	Equipment low Tx power
EQPT	Equipment low transceiver voltage
EQPT	Equipment Rx locked
EQPT	Equipment squelched
EQPT	Equipment Tx locked
BER_SF	OTUk—signal failure
BER_SF	ODUk—signal failure
BER_SD	OTUk—signal degrade
BER_SD	ODUk—signal degrade
UNIDENTIFIED	Pluggable port missing
UNIDENTIFIED	Pluggable port rate mismatch
SECURITY_VIOLATION	Pluggable port security code mismatch
UNIDENTIFIED	TCI not selected
UNIDENTIFIED	TCI 1 clock failure
UNIDENTIFIED	ODUk—backward defect indicator
UNIDENTIFIED	ODUk—open connection indication
UNIDENTIFIED	ODUk—locked defect—[TCM1 | TCM2]
AIS	ODUk—alarm indication signal
TIM	ODUk—trail trace identifier mismatch
BER_SD	ODUk—signal degrade—[TCM1 | TCM2]
BER_SF	ODUk—signal failure—[TCM1 | TCM2]
UNIDENTIFIED	ODUk—open connection indication
UNIDENTIFIED	ODUk—locked defect—TCM1
AIS	ODUk—alarm indication signal
TIM	ODUk—trail trace identifier mismatch
BER_SD	ODUk—signal degrade—TCM1
BER_SF	ODUk—signal failure—TCM1

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
LOF	OTUk—loss of frame
UNIDENTIFIED	TCI 2 clock failure
UNIDENTIFIED	Audit log 80 percent full
UNIDENTIFIED	Module communication failure
UNIDENTIFIED	Need to save running config
UNIDENTIFIED	Audit log 100 percent full; oldest records will be lost
UNIDENTIFIED	Standby database out of sync
EQPT	Redundant power capability lost
EQPT	Equipment wavelength mismatch
UNIDENTIFIED	DSP communication failure
UNIDENTIFIED	DSP failure
UNIDENTIFIED	Loss of channel
UNIDENTIFIED	FEC mismatch
UNIDENTIFIED	Optical power degrade low
UNIDENTIFIED	Optical power degrade high
SECURITY_VIOLATION	Security intrusion attempt detected—see audit log
SECURITY_VIOLATION	Security intrusion attempt detected—see audit log
UNIDENTIFIED	Variable optical attenuator degrade low
UNIDENTIFIED	Variable optical attenuator degrade high
UNIDENTIFIED	Variable optical attenuator failure low
UNIDENTIFIED	Variable optical attenuator failure high
UNIDENTIFIED	Laser bias degrade
UNIDENTIFIED	Laser bias failure
UNIDENTIFIED	Laser temperature degrade
UNIDENTIFIED	Optical amplifier gain degrade low
UNIDENTIFIED	Optical amplifier gain degrade high
UNIDENTIFIED	Optical amplifier gain failure low
UNIDENTIFIED	Optical amplifier gain failure high
UNIDENTIFIED	Laser auto power reduction
UNIDENTIFIED	Case temperature degrade
UNIDENTIFIED	Fiber temperature degrade
UNIDENTIFIED	Shutter open
UNIDENTIFIED	AWG temperature degrade
UNIDENTIFIED	AWG temperature failure
UNIDENTIFIED	AWG over temperature

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Shutter insertion loss variation degrade low
UNIDENTIFIED	Shutter insertion loss variation degrade high
UNIDENTIFIED	AWG warm up
UNIDENTIFIED	Plug-in module communication failure
UNIDENTIFIED	Optical amplifier initialization
UNIDENTIFIED	Optical channel connection failure
UNIDENTIFIED	Optical channel activation failure
UNIDENTIFIED	Optical channel deactivation failure
UNIDENTIFIED	Network topology incomplete
UNIDENTIFIED	Optical network type mismatch
UNIDENTIFIED	Automatic power control failure
UNIDENTIFIED	Automatic power control disabled
UNIDENTIFIED	Ring ID mismatch
TIM	TIM section monitor—trace identifier mismatch failure
UNIDENTIFIED	Optical channel incomplete
UNIDENTIFIED	Enhanced remote failure indication—path—server
UNIDENTIFIED	Enhanced remote failure indication—path—connectivity
UNIDENTIFIED	Enhanced remote failure indication—path—payload
UNIDENTIFIED	Automatic laser shutdown
UNIDENTIFIED	Firewall has been disabled
UNIDENTIFIED	Optical power failure low
UNIDENTIFIED	Optical power failure high
EQPT	Battery failure
EQPT	Extreme high voltage
EQPT	Extreme low voltage
UNIDENTIFIED	High voltage
UNIDENTIFIED	Low voltage
UNIDENTIFIED	Connection equipment mismatch
UNIDENTIFIED	Disable inactive user
UNIDENTIFIED	Disable inactive user
UNIDENTIFIED	Suspend user
UNIDENTIFIED	Suspend user clear
UNIDENTIFIED	Line DCC termination failure
UNIDENTIFIED	Multiplex section DCC termination failure
UNIDENTIFIED	Gigabit Ethernet out of sync

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
AIS	Alarm indication signal in Tx
AIS	Remote alarm indication in Tx
LOF	Loss of frame in Tx
LOS	Incoming signal loss on fibre channel interface
UNIDENTIFIED	Incoming synchronization loss on fibre channel interface
UNIDENTIFIED	Out of frame detected by GFP receiver
UNIDENTIFIED	Client signal loss frames detected by GFP receiver
UNIDENTIFIED	Client synchronization loss frames detected by GFP receiver
UNIDENTIFIED	Sequence mismatch
UNIDENTIFIED	Loss of alignment
UNIDENTIFIED	Out of use—administrative command
UNIDENTIFIED	Out of use—transport failure
UNIDENTIFIED	VCAT group down
UNIDENTIFIED	VCAT group degraded
UNIDENTIFIED	VCAT group incomplete
UNIDENTIFIED	K byte channel failure
UNIDENTIFIED	Industrial high temperature
FOP_APS	APS invalid mode
UNIDENTIFIED	IP address already in use within the same DCC area
UNIDENTIFIED	Node name already in use within the same DCC area
UNIDENTIFIED	Rear panel Ethernet link removed
UNIDENTIFIED	Manual switch to protect resulted in no traffic switch
UNIDENTIFIED	Manual switch back to working resulted in no traffic switch
UNIDENTIFIED	Forced switch to protection resulted in no traffic switch
UNIDENTIFIED	Forced switch back to working resulted in no traffic switch
LOS	Incoming payload signal absent
UNIDENTIFIED	Incoming overhead signal absent
UNIDENTIFIED	Optical safety remote interlock on
UNIDENTIFIED	OSPF hello fail
UNIDENTIFIED	Automatic power control correction skipped
UNIDENTIFIED	Plug-in module range settings mismatch
UNIDENTIFIED	APC cannot set value due to range limits
UNIDENTIFIED	Automatic WDM ANS finished
UNIDENTIFIED	Port failure
UNIDENTIFIED	Unreachable port target power

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Port add power degrade low
UNIDENTIFIED	Port add power degrade high
UNIDENTIFIED	Port add power fail low
UNIDENTIFIED	Port add power fail high
UNIDENTIFIED	Equipment power failure at connector A
UNIDENTIFIED	Equipment power failure at connector B
UNIDENTIFIED	Equipment power failure at return connector A
UNIDENTIFIED	Far-end manual switch back to working—span
UNIDENTIFIED	Far-end forced switch back to working—span
UNIDENTIFIED	Universal transponder module hardware failure
UNIDENTIFIED	Universal transponder module communication failure
UNIDENTIFIED	Automatic power control terminated on manual request
UNIDENTIFIED	Fibre channel distance extension credit starvation
PLM	GFP user payload mismatch
UNIDENTIFIED	GFP fibre channel distance extension mismatch
UNIDENTIFIED	GFP fibre channel distance extension buffer starvation
BER_SF	GFP client signal fail detected
LOF	GFP loss of frame delineation
UNIDENTIFIED	GFP extension header mismatch
UNIDENTIFIED	Signal loss on data interface
UNIDENTIFIED	Encapsulation mismatch—path
UNIDENTIFIED	Encapsulation mismatch—VT
UNIDENTIFIED	Encapsulation mismatch high order—path
UNIDENTIFIED	Encapsulation mismatch low order—path
UNIDENTIFIED	Synchronization loss on data interface
AIS	ODUk-1...4—alarm indication signal
UNIDENTIFIED	LCAS control word CRC check failure
UNIDENTIFIED	Duplicate serial number detected on a pluggable entity
UNIDENTIFIED	LCAS VCG member Tx side in DNU state
UNIDENTIFIED	LCAS VCG member Tx side in ADD state
UNIDENTIFIED	LCAS VCG member Rx side in FAIL state
UNIDENTIFIED	Optimized 1+1 APS primary facility
UNIDENTIFIED	Optimized 1+1 APS primary section mismatch
UNIDENTIFIED	Optimized 1+1 APS invalid primary section
UNIDENTIFIED	Composite clock high line voltage

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
BER_SD	BER threshold exceeded for signal degrade—VT
BER_SF	BER threshold exceeded for signal failure—VT
UNIDENTIFIED	Alarms suppressed on out-of-group VCAT member
UNIDENTIFIED	Span length out of range
UNIDENTIFIED	Temperature reading mismatch between SC cards
UNIDENTIFIED	Voltage reading mismatch between SC cards
TIM	VT path trace identifier mismatch
UNIDENTIFIED	BLSR software version mismatch
UNIDENTIFIED	Bridge and roll occurred
UNIDENTIFIED	Bridge and roll is pending a valid signal
TIM	Open I/O slot(s)
UNIDENTIFIED	ISIS adjacency failure
SECURITY_VIOLATION	Session time limit expired
SECURITY_VIOLATION	User password change required
SECURITY_VIOLATION	Remote authentication fail
TIM	Section trace identifier mismatch
TIM	Regenerator section trace identifier mismatch
UNIDENTIFIED	Switching matrix module failure—working
UNIDENTIFIED	Switching matrix module failure—protect
UNIDENTIFIED	Provisioning mismatch
UNIDENTIFIED	Slot communication disabled
UNIDENTIFIED	MSSP software version mismatch
UNIDENTIFIED	Archival of audit log failed
UNIDENTIFIED	RPR wrapped
UNIDENTIFIED	Software mismatch
UNIDENTIFIED	Link layer keepalive failure
UNIDENTIFIED	Bad packet count exceeds threshold
UNIDENTIFIED	Autonegotiation remote failure indication
UNIDENTIFIED	Shelf communication failure
UNIDENTIFIED	Duplicated shelf identifier
UNIDENTIFIED	Ring is squelching STS traffic
UNIDENTIFIED	Ring is squelching VT traffic
UNIDENTIFIED	Clock bus failure—shelf controller A
UNIDENTIFIED	Clock bus failure—shelf controller B
UNIDENTIFIED	Loss of clock from mate shelf controller

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Payload missing indication
UNIDENTIFIED	Forward defect indication
UNIDENTIFIED	Optical termination incomplete
UNIDENTIFIED	Span loss not checked
UNIDENTIFIED	Ring is squelching higher order traffic
UNIDENTIFIED	Ring is squelching lower order traffic
UNIDENTIFIED	Trail signal fail
UNIDENTIFIED	DS1 loopback command sent to far end
UNIDENTIFIED	Multiplex section—signal degraded
UNIDENTIFIED	Multiplex section—excessive errors
UNIDENTIFIED	High-order path—signal degraded
UNIDENTIFIED	High-order path—excessive errors
UNIDENTIFIED	Low-order path—signal degraded
UNIDENTIFIED	Low-order path—excessive errors
UNIDENTIFIED	Regenerator section—DCC termination failure
UNIDENTIFIED	Network memory pool low
UNIDENTIFIED	OSPF routing table overflow
UNIDENTIFIED	Auto laser shutdown disabled
UNIDENTIFIED	RPR protection is active
UNIDENTIFIED	Max RPR station number exceeded
UNIDENTIFIED	RPR protection configuration mismatched
UNIDENTIFIED	Reserved bandwidth link rate exceeded on ringlet [0|1]
UNIDENTIFIED	RPR interface in pass-through mode
UNIDENTIFIED	RPR peer node is missing
UNIDENTIFIED	RPR RI failure
UNIDENTIFIED	RPR signal failure
UNIDENTIFIED	RPR signal degrade
UNIDENTIFIED	RPR span mismatch
UNIDENTIFIED	LMP failure
UNIDENTIFIED	LMP signal degrade
UNIDENTIFIED	LMP signal failure
UNIDENTIFIED	LMP unallocated data link
UNIDENTIFIED	APC wrong gain setpoint
EQPT	Non-Cisco PPM inserted
EQPT	Unqualified PPM inserted

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Fast automatic protection switching
UNIDENTIFIED	Fast automatic protection switching config mismatch
UNIDENTIFIED	ADM peer cards interlink failure
EQPT	Fan tray mismatch
UNIDENTIFIED	Fibre channel distance extension function not established
UNIDENTIFIED	LCAS sink group error
UNIDENTIFIED	LCAS VCG member Rx side in DNU state
UNIDENTIFIED	Card/ports unable to provide protection
UNIDENTIFIED	TCC front port link loss
UNIDENTIFIED	Shutter insertion loss variation degrade low
UNIDENTIFIED	Shutter insertion loss variation degrade high
UNIDENTIFIED	Idle signal condition
UNIDENTIFIED	Idle signal condition in TX
UNIDENTIFIED	Automatic power control correction skipped
UNIDENTIFIED	Port add power fail low
UNIDENTIFIED	Port add power fail high
UNIDENTIFIED	Span length out of range
UNIDENTIFIED	Port add power degrade low
UNIDENTIFIED	Port add power degrade high
UNIDENTIFIED	DCU loss failure
UNIDENTIFIED	OCHNC maintenance
UNIDENTIFIED	K-byte channel failure
UNIDENTIFIED	PRP protection configuration mismatched
UNIDENTIFIED	Reserved bandwidth link rate exceeded on ring Let0
UNIDENTIFIED	Reserved bandwidth link rate exceeded on ring Let1
UNIDENTIFIED	Security invalid login—locked out
UNIDENTIFIED	Security invalid login—already logged on
UNIDENTIFIED	Syslog messages
UNIDENTIFIED	Configuration events
UNIDENTIFIED	Link up/link down trap
UNIDENTIFIED	Cold/warm start trap
UNIDENTIFIED	Authentication failure notification
EQPT	Unqualified PPM inserted
UNIDENTIFIED	Fast automatic protection switching
UNIDENTIFIED	Fast automatic protection switching config mismatch

Table A-22 CTM GateWay/CORBA Mapping for the ONS 15310 CA, ONS 15310 MA, ONS 15310 MA SDH, ONS 15327 SONET, ONS 15454 SONET, ONS 15454 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	ADM peer cards interlink failure
EQPT	Fan tray mismatch
UNIDENTIFIED	Fibre channel distance extension function not established
UNIDENTIFIED	LCAS sink group error
UNIDENTIFIED	LCAS VCG member Rx side in DNU state
UNIDENTIFIED	Card/ports unable to provide protection
UNIDENTIFIED	TCC front port link loss
UNIDENTIFIED	Shutter insertion loss variation degrade low
UNIDENTIFIED	Shutter insertion loss variation degrade high
UNIDENTIFIED	Idle signal condition
UNIDENTIFIED	Idle signal condition in TX
UNIDENTIFIED	Automatic power control correction skipped
UNIDENTIFIED	Port add power fail low
UNIDENTIFIED	Port add power fail high
UNIDENTIFIED	DCU loss failure
UNIDENTIFIED	OCHNC maintenance
UNIDENTIFIED	K-byte channel failure
UNIDENTIFIED	PRP protection configuration mismatched
UNIDENTIFIED	Reserved bandwidth link rate exceeded on ring Let0
UNIDENTIFIED	Reserved bandwidth link rate exceeded on ring Let1
UNIDENTIFIED	Security invalid login—locked out
UNIDENTIFIED	Security invalid login—already logged on
UNIDENTIFIED	Syslog messages
UNIDENTIFIED	Configuration events
UNIDENTIFIED	Link up/link down trap
UNIDENTIFIED	Cold/warm start trap
UNIDENTIFIED	Authentication failure notification
UNIDENTIFIED	Alarm indication signal in TX—customer installation
UNIDENTIFIED	Remote alarm indication in TX—customer installation

A.5.7 CTM GateWay/CORBA Mapping—ONS 15454-M2 and ONS 15454-M6

The following table lists the CTM GateWay/CORBA mapping for the ONS 15454-M2 and ONS 15454-M6.

Table A-23 CTM GateWay/CORBA Mapping for the ONS 15454-M2 and ONS 15454-M6

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Ethernet OSC termination failure
UNIDENTIFIED	Laser shutdown due to insertion of non-Cisco PPM
UNIDENTIFIED	Software signature verification failure
UNIDENTIFIED	Active volume software sgnature verification failure
UNIDENTIFIED	Protect volume software signature verification failure
UNIDENTIFIED	Channel shutdown due to wavelength drift
UNIDENTIFIED	Software activation failure
UNIDENTIFIED	USB write failure
UNIDENTIFIED	USB sync in progress

A.5.8 CTM GateWay/CORBA Mapping—ONS 15530

The following table lists the CTM GateWay/CORBA mapping for the ONS 15530.

Table A-24 CTM GateWay/CORBA Mapping for the ONS 15530

TMF Mapping	NE Native Probable Cause
FAN Alarms	
EQPT	Two or more fans failed
EQPT	One fan failed
CPU Alarms	
UNIDENTIFIED	Unknown alarm detected
EQPT	SRC diag failure
EQPT	PCI diag failure
EQPT	PCMCIA diag failure
EQPT	Cannot access multiple line card
EQPT	IDPROM/backplane slot mismatch
EQPT	Cannot access backplane IDPROM
EQPT	Inconsistent redun states
EQPT	Read invalid SRC
EQPT	Unprotected; peer missing
EQPT	Unprotected; peer not responding
EQPT	Standby with lower capabilities

Table A-24 CTM GateWay/CORBA Mapping for the ONS 15530 (continued)

TMF Mapping	NE Native Probable Cause
TSP Alarms	
EQPT	Access to TSP card failed
EQPT	Access to IDPROM failed
TX_FAIL	Line laser failure detected
TX_DEGRADE	Trunk laser degrade detected
UNIDENTIFIED	Trunk laser lambda deviation
UNIDENTIFIED	Wavelength mismatch error
MDSUBCARD Alarms	
EQPT	Access to IDPROM failed
Client Port Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
LOF	SONET loss of frame event
UNIDENTIFIED	SONET severely errored frame
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by open fibre control
Wave Port Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
LOF	SONET loss of frame event
UNIDENTIFIED	SONET severely errored frame
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by laser safety control
UNIDENTIFIED	Link down
Chassis Alarm	
EQPT	Chassis fan tray missing
ENV	Chassis temp > critical limit
ENV	Chassis temp > major limit
ENV	Chassis temp > minor limit
ENV	Chassis temp less than –5 C
ENV	Chassis power supply 0 missing
ENV	Chassis power supply 1 missing
Power Supply Alarms	
EQPT	Power supply unit failed

Table A-24 CTM GateWay/CORBA Mapping for the ONS 15530 (continued)

TMF Mapping	NE Native Probable Cause
Wave OSC Alarms	
UNIDENTIFIED	Loss of lock event
OSC_LOS	Loss of signal event
Wave Ethernet Phy Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by laser safety control
Ten Gig Ethernet Phy Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by laser safety control
ESCON Port Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
UNIDENTIFIED	Laser disabled event
UNIDENTIFIED	Local failure, laser disabled
ESCON SFP Alarms	
TX_FAIL	Transmit laser failed
10G Card Alarms	
EQPT	Access to 10G card failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
EQPT	10G internal loopback failed
EQPT	10G internal loopback through PSC failed
TX_FAIL	Laser failure detected
UNIDENTIFIED	Laser wavelength deviation
ESCON Card Alarms	
EQPT	Access to ESCON card failed
EQPT	Access to IDPROM failed
EQPT	ESCON card internal loopback failed
EQPT	ESCON card internal loopback through PSC failed
OSC MB Alarms	

Table A-24 CTM GateWay/CORBA Mapping for the ONS 15530 (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Access to card failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
OSC Subcard Alarms	
EQPT	Access to card failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Ethernet DCC loopback from OSCMB failed
TX_FAIL	Laser failure detected
UNIDENTIFIED	Laser wavelength deviation
VOA and PEQ SubCard Alarms	
EQPT	Access to card failed
EQPT	Access to IDPROM failed
IF-THRESHOLD Alarms	
UNIDENTIFIED	CVRD failure threshold exceeded
UNIDENTIFIED	CVRD degrade threshold exceeded
UNIDENTIFIED	CDL HEC failure threshold exceeded
UNIDENTIFIED	CDL HEC degrade threshold exceeded
UNIDENTIFIED	SONET/SDH CV-S failure threshold exceeded
UNIDENTIFIED	SONET/SDH CV-S degrade threshold exceeded
UNIDENTIFIED	CDL header CRC failure threshold exceeded
UNIDENTIFIED	CDL header CRC degrade threshold exceeded
UNIDENTIFIED	ESCON encapsulation packet errors failure threshold exceeded
UNIDENTIFIED	ESCON encapsulation packet errors degrade threshold exceeded
CDL Alarms	
UNIDENTIFIED	CDL hop-by-hop backward defect indication
UNIDENTIFIED	CDL hop-by-hop forward defect indication
UNIDENTIFIED	CDL end-by-end forward defect indication
UNIDENTIFIED	CDL end-by-end backward defect indication
Transceiver Alarms	
TX_FAIL	Laser failure
2.5G Card Alarms	
EQPT	Access to 2.5G card failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
EQPT	2.5G internal loopback failed
EQPT	2.5G internal loopback through PSC failed

Table A-24 CTM GateWay/CORBA Mapping for the ONS 15530 (continued)

TMF Mapping	NE Native Probable Cause
TX_FAIL	Laser failure detected
UNIDENTIFIED	Laser wavelength deviation
FCGE Card Alarms	
EQPT	Access to card failed
EQPT	Access to IDPROM failed
EQPT	Card internal loopback failed
EQPT	Card internal loopback through PSC failed

A.5.9 CTM GateWay/CORBA Mapping—ONS 15540

The following table lists the CTM GateWay/CORBA mapping for the ONS 15540.

Table A-25 CTM GateWay/CORBA Mapping for the ONS 15540

TMF Mapping	NE Native Probable Cause
Chassis Alarms	
EQPT	Chassis fan tray missing
ENV	Chassis temp > critical limit
ENV	Chassis temp > major limit
ENV	Chassis temp > minor limit
ENV	Chassis temp less than –15 C
ENV	Chassis power supply A failed
ENV	Chassis power supply B failed
FAN Alarms	
EQPT	Two or more fans failed
EQPT	One fan failed
CPU Alarms	
UNIDENTIFIED	Unknown alarm detected
EQPT	SRC diag failure
EQPT	PCI diag failure
EQPT	PCMCIA diag failure
EQPT	Cannot access multiple line card
EQPT	IDPROM/backplane slot mismatch
EQPT	Cannot access backplane IDPROM
EQPT	Inconsistent redun states
EQPT	Read invalid SRC
EQPT	Unprotected; peer missing
EQPT	Unprotected; peer not responding

Table A-25 CTM GateWay/CORBA Mapping for the ONS 15540 (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Standby with lower capabilities
LCMB Alarms	
EQPT	Access to LRC failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Waveport [0...3] opt switch failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
MDMBOSC Alarms	
EQPT	Access to LRC failed
EQPT	Access to IDPROM failed
EQPT	OSC hardware failure detected
TX_FAIL	Trunk laser failure detected
UNIDENTIFIED	Trunk laser lambda deviation
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
MDMB Alarms	
EQPT	Access to LRC failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
TSP1 Alarms	
EQPT	Access to TSP card failed
EQPT	Access to IDPROM failed
TX_FAIL	Line laser failure detected
TX_FAIL	Trunk laser failure detected
UNIDENTIFIED	Trunk laser lambda deviation
UNIDENTIFIED	Wavelength mismatch error
MD Subcard Alarms	
EQPT	Access to IDPROM failed
Client Port Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
LOF	SONET loss of frame event
UNIDENTIFIED	SONET severely errored frame
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by open fibre control
Wave Port Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event

Table A-25 CTM GateWay/CORBA Mapping for the ONS 15540 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Loss of sync event
LOF	SONET loss of frame event
UNIDENTIFIED	SONET severely errored frame
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by laser safety control
UNIDENTIFIED	Link down
IF-THRESHOLD Alarms	
UNIDENTIFIED	CVRD failure threshold exceeded
UNIDENTIFIED	CVRD degrade threshold exceeded
UNIDENTIFIED	HEC failure threshold exceeded
UNIDENTIFIED	HEC degrade threshold exceeded
UNIDENTIFIED	SONET/SDH CV-S failure threshold exceeded
UNIDENTIFIED	SONET/SDH CV-S degrade threshold exceeded
CDL Alarms	
UNIDENTIFIED	CDL hop-by-hop backward defect indication
UNIDENTIFIED	CDL hop-by-hop forward defect indication
UNIDENTIFIED	CDL end-by-end forward defect indication
UNIDENTIFIED	CDL end-by-end backward defect indication
Wave Ethernet Phy Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event
UNIDENTIFIED	Loss of sync event
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Laser shut down by laser safety control
Ten Gig Ethernet Phy Alarms	
UNIDENTIFIED	Loss of lock event
LOS	Loss of signal event in Cisco OPT-MON-MIB
UNIDENTIFIED	Loss of sync event
UNIDENTIFIED	Laser shut down by forward laser control
UNIDENTIFIED	Trunk laser shut down by laser safety control
10G Card Alarms	
EQPT	Access to card failed
EQPT	Access to IDPROM failed
EQPT	10G internal loopback failed
TX_FAIL	Line laser failed
TX_FAIL	Trunk laser failure

Table A-25 CTM GateWay/CORBA Mapping for the ONS 15540 (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Trunk laser wavelength deviation
LOF	GFP loss of frame delineation
UNIDENTIFIED	Transport layer failure
UNIDENTIFIED	Encapsulation mismatch path
10G MB Alarms	
EQPT	Access to LRC failed
EQPT	Access to IDPROM failed
UNIDENTIFIED	Waveport [0...3] OPT switch failed
UNIDENTIFIED	Ethernet DCC loopback from PSC failed
Dummy TSP for 10GE Alarms	
EQPT	Access to subcard failed
EQPT	Access to IDPROM failed
TSP2 Alarms	
EQPT	Access to TSP2 card failed
EQPT	Access to IDPROM failed
TX_FAIL	Line laser failure
TX_FAIL	Trunk laser failure detected
UNIDENTIFIED	Trunk laser lambda deviation
Transceiver Alarms	
TX_FAIL	Laser failure

A.5.10 Optical Monitoring Alarms—ONS 15530 and ONS 15540

The following table lists the optical monitoring alarms applicable to both the ONS 15530 and ONS 15540.

Table A-26 Optical Monitoring Alarms for the ONS 15530, ONS 15540

TMF Mapping	NE Native Probable Cause
Power Optical Monitor Alarms	
UNIDENTIFIED	Rx optical power exceeds high alarm threshold
UNIDENTIFIED	Tx optical power exceeds high alarm threshold
UNIDENTIFIED	Rx optical power exceeds high warning threshold
UNIDENTIFIED	Tx optical power exceeds high warning threshold
UNIDENTIFIED	Rx optical power below low alarm threshold
UNIDENTIFIED	Tx optical power exceeds high alarm threshold
UNIDENTIFIED	Rx optical power below low warning threshold
UNIDENTIFIED	Tx optical power below low warning threshold

Table A-26 Optical Monitoring Alarms for the ONS 15530, ONS 15540 (continued)

TMF Mapping	NE Native Probable Cause
AC Power Optical Monitor Alarms	
UNIDENTIFIED	Rx optical AC power exceeds high alarm threshold
UNIDENTIFIED	Tx optical AC power exceeds high alarm threshold
UNIDENTIFIED	Rx optical AC power exceeds high warning threshold
UNIDENTIFIED	Tx optical AC power exceeds high warning threshold
UNIDENTIFIED	Rx optical AC power below low alarm threshold
UNIDENTIFIED	Tx optical AC power below low alarm threshold
UNIDENTIFIED	Rx optical AC power below low warning threshold
UNIDENTIFIED	Tx optical AC power below low warning threshold
Ambient Temp Power Optical Monitor Alarms	
UNIDENTIFIED	Rx ambient temp exceeds high alarm threshold
UNIDENTIFIED	Tx ambient temp exceeds high alarm threshold
UNIDENTIFIED	Rx ambient temp exceeds high warning threshold
UNIDENTIFIED	Tx ambient temp exceeds high warning threshold
UNIDENTIFIED	Rx ambient temp below low alarm threshold
UNIDENTIFIED	Tx ambient temp below low alarm threshold
UNIDENTIFIED	Rx ambient temp below low warning threshold
UNIDENTIFIED	Tx ambient temp below low warning threshold
Laser Temp Power Optical Monitor Alarms	
UNIDENTIFIED	Rx laser temp exceeds high alarm threshold
UNIDENTIFIED	Tx laser temp exceeds high alarm threshold
UNIDENTIFIED	Rx laser temp exceeds high warning threshold
UNIDENTIFIED	Tx laser temp exceeds high warning threshold
UNIDENTIFIED	Rx laser temp below low alarm threshold
UNIDENTIFIED	Tx laser temp below low alarm threshold
UNIDENTIFIED	Rx laser temp below low warning threshold
UNIDENTIFIED	Tx laser temp below low warning threshold
Bias Temp Monitor Alarms	
UNIDENTIFIED	Rx bias current exceeds high alarm threshold
UNIDENTIFIED	Tx bias current exceeds high alarm threshold
UNIDENTIFIED	Rx bias current exceeds high warning threshold
UNIDENTIFIED	Tx bias current exceeds high warning threshold
UNIDENTIFIED	Rx bias current below low alarm threshold
UNIDENTIFIED	Tx bias current below low alarm threshold
UNIDENTIFIED	Rx bias current below low warning threshold
UNIDENTIFIED	Tx bias current below low warning threshold

Table A-26 Optical Monitoring Alarms for the ONS 15530, ONS 15540 (continued)

TMF Mapping	NE Native Probable Cause
Laser Peltier Current Monitor Alarms	
UNIDENTIFIED	Rx laser Peltier current exceeds high alarm threshold
UNIDENTIFIED	Tx laser Peltier current exceeds high alarm threshold
UNIDENTIFIED	Rx laser Peltier current exceeds high warning threshold
UNIDENTIFIED	Tx laser Peltier current exceeds high warning threshold
UNIDENTIFIED	Rx laser Peltier current below low alarm threshold
UNIDENTIFIED	Tx laser Peltier current below low alarm threshold
UNIDENTIFIED	Rx laser Peltier current below low warning threshold
UNIDENTIFIED	Tx laser Peltier current below low warning threshold
Transceiver Voltage Monitor Alarms	
UNIDENTIFIED	Rx transceiver voltage exceeds high alarm threshold
UNIDENTIFIED	Tx transceiver voltage exceeds high alarm threshold
UNIDENTIFIED	Rx transceiver voltage exceeds high warning threshold
UNIDENTIFIED	Tx transceiver voltage exceeds high warning threshold
UNIDENTIFIED	Rx transceiver voltage below low alarm threshold
UNIDENTIFIED	Tx transceiver voltage below low alarm threshold
UNIDENTIFIED	Rx transceiver voltage below low warning threshold
UNIDENTIFIED	Tx transceiver voltage below low warning threshold

A.5.11 CTM GateWay/CORBA Mapping—ONS 15600 SONET

The following table lists the CTM GateWay/CORBA mapping for the ONS 15600 SONET.

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET

TMF Mapping	NE Native Probable Cause
EQPT	Alarms/events suppressed for this object
UNIDENTIFIED	Audit log 100 percent full
UNIDENTIFIED	Audit log 80 percent full
UNIDENTIFIED	BLSR out of sync
UNIDENTIFIED	BLSR multinode table update completed
UNIDENTIFIED	TSC switched to alternate Ethernet port
ENV	Failure detected external to the NE
DCC_FAILURE	DCC channel loss
DCC_FAILURE	SDCC termination failure
AIS	Alarm indication signal
AIS	Alarm indication signal—line
AIS	Alarm indication signal—path

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
FOP_APS	Byte failure
FOP_APS	Protection switching channel match failure
FOP_APS	Automatic protection switch mode mismatch
FOP_APS	Connection loss
FOP_APS	Default K byte
FOP_APS	Far-end protection line failure
FOP_APS	APS channel—far-end protection line signal degrade
FOP_APS	Improper APS code
FOP_APS	Inconsistent APS code
FOP_APS	Node ID mismatch
UNIDENTIFIED	STS concatenation error
LOF	Loss of frame
LOP	Loss of pointer—path
LOS	Loss of signal
RAI	Remote failure indication—line
RAI	Remote failure indication—path
BER_SD	BER threshold exceeded for signal degrade—line
BER_SD	BER threshold exceeded for signal degrade—path
BER_SD	Far-end BER threshold passed for signal degrade—ring
BER_SD	Far-end BER threshold passed for signal degrade—span
BER_SF	BER threshold exceeded for signal failure—line
BER_SF	BER threshold exceeded for signal failure—path
BER_SF	Far-end BER threshold passed for signal failure—ring
BER_SF	Far-end BER threshold passed for signal failure—span
UNIDENTIFIED	Payload defect indication—path
UNEQ	Unequipped—path
UNIDENTIFIED	Don’t use for synchronization
UNIDENTIFIED	Failed to receive synchronization status message
UNIDENTIFIED	Synchronization status messages are disabled on this interface
UNIDENTIFIED	Primary reference source—stratum 1 traceable
UNIDENTIFIED	Reserved for network synchronization use
UNIDENTIFIED	SONET minimum clock traceable
UNIDENTIFIED	Stratum 2 traceable
UNIDENTIFIED	Stratum 3 traceable
UNIDENTIFIED	Stratum 3E traceable
UNIDENTIFIED	Stratum 4 traceable

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Synchronized—traceability unknown
UNIDENTIFIED	Transit node clock traceable
UNIDENTIFIED	Fifth synchronization reference failure
UNIDENTIFIED	Fourth synchronization reference failure
UNIDENTIFIED	Synchronization reference frequency out of bounds
UNIDENTIFIED	Primary synchronization reference failure
UNIDENTIFIED	Secondary synchronization reference failure
UNIDENTIFIED	Sixth synchronization reference failure
UNIDENTIFIED	Third synchronization reference failure
TIM	STS path trace identifier mismatch
ENV	NE power failure at connector A
ENV	NE power failure at connector B
EQPT	Power fuse failure
UNIDENTIFIED	Free running synchronization mode
UNIDENTIFIED	Stratum 3E fast start synchronization mode
UNIDENTIFIED	Holdover synchronization mode
UNIDENTIFIED	Autonomous messages inhibited
UNIDENTIFIED	Autonomous PM report message inhibited
UNIDENTIFIED	Mismatch of equipment and attributes
UNIDENTIFIED	Ring topology is under construction
ENV	Software fault—data integrity fault
ENV	Airflow failure
EQPT	Equipment fails to boot
UNIDENTIFIED	Connection ID mismatch on CXC 0
UNIDENTIFIED	Connection ID mismatch on CXC 1
EQPT	Clock module failure
EQPT	Control bus failure—I/O—TSC A
EQPT	Control bus failure—I/O—TSC B
EQPT	Clock bus failure—TSC A
EQPT	Clock bus failure—TSC B
EQPT	Control communication equipment failure
EQPT	Primary nonvolatile backup memory failure
EQPT	Manufacturing data memory (EEPROM) failure
EQPT	Payload bus failure—matrix A
EQPT	Payload bus failure—matrix B
UNIDENTIFIED	CXC operations suspended

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Equipment failure
EQPT	Diagnostics failure
EQPT	Equipment failure—high temperature
EQPT	Invalid MAC address
UNIDENTIFIED	Failure to switch to protection
UNIDENTIFIED	Failure to switch to protection—path
UNIDENTIFIED	Failure to switch to protection—ring
UNIDENTIFIED	Failure to switch to protection—span
EQPT	Fan failure
EQPT	Partial fan failure—speed degradation
EQPT	Partial fan failure—parts failure
EQPT	Clock module frequency mismatch
UNIDENTIFIED	OSPF hello fail
EQPT	Laser bias current high
EQPT	High laser temperature
UNIDENTIFIED	Software download failed
UNIDENTIFIED	Different software version
EQPT	Synchronization equipment unavailable
EQPT	Unprotected synchronization equipment
EQPT	Unprotected matrix equipment
UNIDENTIFIED	System upgrade in progress
EQPT	Matrix equipment unavailable
UNIDENTIFIED	Loopback—cross connect
UNIDENTIFIED	Loopback facility
UNIDENTIFIED	Loopback payload
UNIDENTIFIED	Loopback
SECURITY_VIOLATION	Security—invalid login (see audit trail)
SECURITY_VIOLATION	Invalid login attempt threshold exceeded
UNIDENTIFIED	Normal condition
UNIDENTIFIED	Performance monitor threshold crossing alert
UNIDENTIFIED	Duplicate node ID
EQPT	Improper removal
UNIDENTIFIED	Both ends of fiber provisioned as east
UNIDENTIFIED	Far end of fiber is provisioned with different ring ID
UNIDENTIFIED	Protection switch
EQPT	Equipment power failure

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
EQPT	Equipment power failure at connector A
EQPT	Equipment power failure at connector B
EQPT	Equipment power failure at return connector A
EQPT	Equipment power failure at return connector B
UNIDENTIFIED	Automatic reset
UNIDENTIFIED	Automatic path-protection switch caused by AIS
UNIDENTIFIED	Automatic path-protection switch caused by LOP
UNIDENTIFIED	Automatic path-protection switch caused by PDI
UNIDENTIFIED	Automatic path-protection switch caused by SDBER
UNIDENTIFIED	Automatic path-protection switch caused by SFBER
UNIDENTIFIED	Automatic path-protection switch caused by UNEQ
UNIDENTIFIED	Cold restart
UNIDENTIFIED	Exercise ring
UNIDENTIFIED	Exercising ring successfully
UNIDENTIFIED	Far-end exercise ring
UNIDENTIFIED	Exercise request on ring failed
UNIDENTIFIED	Exercise span
UNIDENTIFIED	Exercising span successfully
UNIDENTIFIED	Far-end exercise span
UNIDENTIFIED	Exercise request on span failed
UNIDENTIFIED	Force switch request on facility/equipment
UNIDENTIFIED	Force switch request on ring
UNIDENTIFIED	Force switch request on span
UNIDENTIFIED	Far-end working facility forced to switch to protection unit—ring
UNIDENTIFIED	Far-end working facility forced to switch to protection unit—span
UNIDENTIFIED	Bidirectional full pass-through is active
UNIDENTIFIED	Unidirectional full pass-through is active
EQPT	Inhibit switch to protect request on equipment
EQPT	Inhibit switch to working request on equipment
UNIDENTIFIED	K bytes pass-through is active
UNIDENTIFIED	Far-end lockout of protection—all spans
UNIDENTIFIED	Far-end lockout of protection—ring
UNIDENTIFIED	Far-end lockout of working—ring
UNIDENTIFIED	Far-end lockout of working—span
UNIDENTIFIED	Far-end lockout of protection—span

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Lockout of protection
UNIDENTIFIED	Lockout of protection—ring
UNIDENTIFIED	Lockout of protection—span
UNIDENTIFIED	Lockout of working—ring
UNIDENTIFIED	Lockout of working—span
UNIDENTIFIED	Lockout switch request on facility/equipment
UNIDENTIFIED	Lockout switch request on ring
UNIDENTIFIED	Manual reset
UNIDENTIFIED	Far-end manual switch of working facility to protection unit—ring
UNIDENTIFIED	Far-end manual switch of working facility to protection unit—span
UNIDENTIFIED	Manual switch to fifth reference
UNIDENTIFIED	Manual switch to fourth reference
UNIDENTIFIED	Manual switch to internal clock
UNIDENTIFIED	Manual switch to primary reference
UNIDENTIFIED	Manual switch to second reference
UNIDENTIFIED	Manual switch to sixth reference
UNIDENTIFIED	Manual switch to third reference
UNIDENTIFIED	Manual switch request on facility/equipment
UNIDENTIFIED	Manual switch request on ring
UNIDENTIFIED	Manual switch request on span
UNIDENTIFIED	Powerfail restart
UNIDENTIFIED	Ring is segmented
UNIDENTIFIED	Ring switch is active on the east side
UNIDENTIFIED	Ring switch is active on the west side
UNIDENTIFIED	Span switch is active on the east side
UNIDENTIFIED	Span switch is active on the west side
UNIDENTIFIED	Ring is squelching traffic
UNIDENTIFIED	Squelching path
UNIDENTIFIED	Software download in progress
UNIDENTIFIED	Switch to fifth reference
UNIDENTIFIED	Switch to fourth reference
UNIDENTIFIED	Switch to primary reference
UNIDENTIFIED	Switch to second reference
UNIDENTIFIED	Switch to sixth reference
UNIDENTIFIED	Switch to third reference

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
ENV	System reboot
UNIDENTIFIED	Extra traffic preempted
UNIDENTIFIED	Switched back to working unit
UNIDENTIFIED	Switched to protection unit
UNIDENTIFIED	Warm restart
UNIDENTIFIED	Wait to restore
UNIDENTIFIED	Ring is in wait-to-restore state
UNIDENTIFIED	Span is in wait-to-restore state
UNIDENTIFIED	Bridge and roll has occurred
UNIDENTIFIED	Bridge and roll is pending a valid signal
UNIDENTIFIED	Admin logout of user
UNIDENTIFIED	Admin lockout of user
UNIDENTIFIED	Admin lockout clear
UNIDENTIFIED	Automatic logout of idle user
UNIDENTIFIED	Login of user
SECURITY_VIOLATION	Invalid login—locked out
SECURITY_VIOLATION	Invalid login—already logged on
SECURITY_VIOLATION	Invalid login—password
SECURITY_VIOLATION	Invalid login—username
UNIDENTIFIED	Logout of user
UNIDENTIFIED	User locked out
UNIDENTIFIED	Open I/O slot(s)
UNIDENTIFIED	Improper matrix equipment inserted
EQPT	Equipment fails to boot
UNIDENTIFIED	No description
UNIDENTIFIED	Firewall has been disabled
UNIDENTIFIED	Loss of clock from mate TSC
UNIDENTIFIED	Alarms suppressed for maintenance
UNIDENTIFIED	SNTP host failure
UNIDENTIFIED	Disable inactive user
UNIDENTIFIED	Disable inactive clear
UNIDENTIFIED	Suspend user
UNIDENTIFIED	Duplicate serial number detected on a pluggable entity
EQPT	Equipment problem on carrier or PIM
EQPT	Equipment problem on PIM or PPM
UNIDENTIFIED	Suspend user clear

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Forced switch to primary reference
UNIDENTIFIED	Forced switch to secondary reference
UNIDENTIFIED	Forced switch to third reference
UNIDENTIFIED	Forced switch to internal clock
PLM	GFP user payload mismatch
UNIDENTIFIED	GFP fibre channel distance extension mismatch
UNIDENTIFIED	GFP fibre channel distance extension buffer starvation
BER_SF	GFP client signal fail detected
LOF	GFP loss of frame delineation
UNIDENTIFIED	GFP extension header mismatch
UNIDENTIFIED	Carrier loss on the LAN
UNIDENTIFIED	Encapsulation type mismatch
UNIDENTIFIED	Transport layer failure
PLM	Payload label mismatch—path
DCC_FAILURE	Line DCC termination failure
UNIDENTIFIED	K byte channel failure
UNIDENTIFIED	BLSR software version mismatch
SECURITY_VIOLATION	Security intrusion attempt detected—see audit log
UNIDENTIFIED	IP address already in use within the same DCC area
UNIDENTIFIED	Node name already in use within the same DCC area
UNIDENTIFIED	Free memory on card near zero
UNIDENTIFIED	Free memory on card very low
ENV	NE power failure at connector
UNIDENTIFIED	Standby database out of sync
UNIDENTIFIED	Database backup failed
UNIDENTIFIED	Database restore failed
UNIDENTIFIED	ISIS adjacency failure
EQPT	Equipment high laser bias
EQPT	Equipment high Rx power
EQPT	Equipment high Tx power
EQPT	Equipment low laser bias
EQPT	Equipment low Rx power
EQPT	Equipment low Tx power
UNIDENTIFIED	Provisioning mismatch
UNIDENTIFIED	Illegal route addition to the network
UNIDENTIFIED	Session time limit expired

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	User password change required
SECURITY_VIOLATION	User authentication rejected
UNIDENTIFIED	APS invalid mode
UNIDENTIFIED	Far-end manual switch back to working—span
UNIDENTIFIED	Far-end forced switch back to working—span
UNIDENTIFIED	Alarms suppressed by user command
UNIDENTIFIED	Ring is squelching STS traffic
UNIDENTIFIED	REPT^DBCHG messages inhibited
TIM	Regenerator section trace identifier mismatch
MS-AIS	AIS—multiplex section—alarm indication signal
RAI	RFI—multiplex section—remote failure/alarm indication
BER_SF	Multiplex section—excessive BER
BER_SD	Multiplex section—signal degrade
DCC_FAILURE	Multiplex section DCC termination failure
UNIDENTIFIED	G811—primary reference clock traceable
UNIDENTIFIED	G812T—transit node clock traceable
UNIDENTIFIED	G812L—local node clock traceable
UNIDENTIFIED	G813—synchronous equipment timing source traceable
UNIDENTIFIED	MS-SPRing out of sync
AU-AIS	AIS—administration unit—alarm indication signal
LOP	LOP—administration unit—loss of pointer
BER_SF	High-order path—excessive BER
BER_SD	High-order path—signal degrade
PLM	SLMF—PLM high order—path label mismatch
TIM	TIM high order—trace identifier mismatch failure
RAI	RFI—high order—remote failure/alarm indication
UNEQ	SLMF—unequipped high order—path unequipped
UNIDENTIFIED	Automatic SNCP switch caused by AIS
UNIDENTIFIED	Automatic SNCP switch caused by LOP
UNIDENTIFIED	Automatic SNCP switch caused by UNEQ
UNIDENTIFIED	Automatic SNCP switch caused by SFBER
UNIDENTIFIED	Automatic SNCP switch caused by SDBER
UNIDENTIFIED	Failure to switch to protection—high-order path
UNIDENTIFIED	MSSP multinode table update completed
UNIDENTIFIED	Bipolar violation
UNIDENTIFIED	High-order path—payload defect indication

Table A-27 CTM GateWay/CORBA Mapping for the ONS 15600 SONET (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	MSSP software version mismatch
DCC_FAILURE	Regenerator section—DCC termination failure
EQPT	Battery failure
UNIDENTIFIED	Extreme high voltage
UNIDENTIFIED	Extreme low voltage
UNIDENTIFIED	High voltage
UNIDENTIFIED	Low voltage
UNIDENTIFIED	Voltage reading mismatch between SC cards
EQPT	Wavelength out of lock
EQPT	Automatic laser shutdown
UNIDENTIFIED	Ring is squelching high-order traffic
EQPT	Non-Cisco PPM inserted
EQPT	Unqualified PPM inserted

A.5.12 CTM GateWay/CORBA Mapping—ONS 15600 SDH

The following table lists the CTM GateWay/CORBA mapping for the ONS 15600 SDH.

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH

TMF Mapping	NE Native Probable Cause
EQPT	Alarms/events suppressed for this object
UNIDENTIFIED	Audit log 100 percent full
UNIDENTIFIED	Audit log 80 percent full
UNIDENTIFIED	BLSR out of sync
UNIDENTIFIED	BLSR multinode table update completed
UNIDENTIFIED	TSC switched to alternate Ethernet port
ENV	Failure detected external to the NE
DCC_FAILURE	DCC channel loss
DCC_FAILURE	SDCC termination failure
AIS	Alarm indication signal
AIS	Alarm indication signal—line
AIS	Alarm indication signal—path
FOP_APS	Byte failure
FOP_APS	Protection switching channel match failure
FOP_APS	Automatic protection switch mode mismatch
FOP_APS	Connection loss

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
FOP_APS	Default K byte
FOP_APS	Far-end protection line failure
FOP_APS	APS channel—far-end protection line signal degrade
FOP_APS	Improper APS code
FOP_APS	Inconsistent APS code
FOP_APS	Node ID mismatch
UNIDENTIFIED	STS concatenation error
LOF	Loss of frame
LOP	Loss of pointer—path
LOS	Loss of signal
RAI	Remote failure indication—line
RAI	Remote failure indication—path
BER_SD	BER threshold exceeded for signal degrade—line
BER_SD	BER threshold exceeded for signal degrade—path
BER_SD	Far-end BER threshold passed for signal degrade—ring
BER_SD	Far-end BER threshold passed for signal degrade—span
BER_SF	BER threshold exceeded for signal failure—line
BER_SF	BER threshold exceeded for signal failure—path
BER_SF	Far-end BER threshold passed for signal failure—ring
BER_SF	Far-end BER threshold passed for signal failure—span
UNIDENTIFIED	Payload defect indication—path
UNEQ	Unequipped—path
UNIDENTIFIED	Don’t use for synchronization
UNIDENTIFIED	Failed to receive synchronization status message
UNIDENTIFIED	Synchronization status messages are disabled on this interface
UNIDENTIFIED	Primary reference source—stratum 1 traceable
UNIDENTIFIED	Reserved for network synchronization use
UNIDENTIFIED	SONET minimum clock traceable
UNIDENTIFIED	Stratum 2 traceable
UNIDENTIFIED	Stratum 3 traceable
UNIDENTIFIED	Stratum 3E traceable
UNIDENTIFIED	Stratum 4 traceable
UNIDENTIFIED	Synchronized—traceability unknown
UNIDENTIFIED	Transit node clock traceable
UNIDENTIFIED	Fifth synchronization reference failure
UNIDENTIFIED	Fourth synchronization reference failure

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Synchronization reference frequency out of bounds
UNIDENTIFIED	Primary synchronization reference failure
UNIDENTIFIED	Secondary synchronization reference failure
UNIDENTIFIED	Sixth synchronization reference failure
UNIDENTIFIED	Third synchronization reference failure
TIM	STS path trace identifier mismatch
ENV	NE power failure at connector A
ENV	NE power failure at connector B
EQPT	Power fuse failure
UNIDENTIFIED	Free running synchronization mode
UNIDENTIFIED	Stratum 3E fast start synchronization mode
UNIDENTIFIED	Holdover synchronization mode
UNIDENTIFIED	Autonomous messages inhibited
UNIDENTIFIED	Autonomous PM report message inhibited
UNIDENTIFIED	Mismatch of equipment and attributes
UNIDENTIFIED	Ring topology is under construction
ENV	Software fault—data integrity fault
ENV	Airflow failure
EQPT	Equipment fails to boot
UNIDENTIFIED	Connection ID mismatch on CXC 0
UNIDENTIFIED	Connection ID mismatch on CXC 1
EQPT	Clock module failure
EQPT	Control bus failure—I/O—TSC A
EQPT	Control bus failure—I/O—TSC B
EQPT	Clock bus failure—TSC A
EQPT	Clock bus failure—TSC B
EQPT	Control communication equipment failure
EQPT	Primary nonvolatile backup memory failure
EQPT	Manufacturing data memory (EEPROM) failure
EQPT	Payload bus failure—matrix A
EQPT	Payload bus failure—matrix B
UNIDENTIFIED	CXC operations suspended
EQPT	Equipment failure
EQPT	Diagnostics failure
EQPT	Equipment failure—high temperature
EQPT	Invalid MAC address

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Failure to switch to protection
UNIDENTIFIED	Failure to switch to protection—path
UNIDENTIFIED	Failure to switch to protection—ring
UNIDENTIFIED	Failure to switch to protection—span
EQPT	Fan failure
EQPT	Partial fan failure—speed degradation
EQPT	Partial fan failure—parts failure
EQPT	Clock module frequency mismatch
UNIDENTIFIED	OSPF hello fail
EQPT	Laser bias current high
EQPT	High laser temperature
UNIDENTIFIED	Software download failed
UNIDENTIFIED	Different software version
EQPT	Synchronization equipment unavailable
EQPT	Unprotected synchronization equipment
EQPT	Unprotected matrix equipment
UNIDENTIFIED	System upgrade in progress
EQPT	Matrix equipment unavailable
UNIDENTIFIED	Loopback—cross connect
UNIDENTIFIED	Loopback facility
UNIDENTIFIED	Loopback payload
UNIDENTIFIED	Loopback
SECURITY_VIOLATION	Security—invalid login (see audit trail)
SECURITY_VIOLATION	Invalid login attempt threshold exceeded
UNIDENTIFIED	Normal condition
UNIDENTIFIED	Performance monitor threshold crossing alert
UNIDENTIFIED	Duplicate node ID
EQPT	Improper removal
UNIDENTIFIED	Both ends of fiber provisioned as east
UNIDENTIFIED	Far end of fiber is provisioned with different ring ID
UNIDENTIFIED	Protection switch
EQPT	Equipment power failure
EQPT	Equipment power failure at connector A
EQPT	Equipment power failure at connector B
EQPT	Equipment power failure at return connector A
EQPT	Equipment power failure at return connector B

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Automatic reset
UNIDENTIFIED	Automatic path-protection switch caused by AIS
UNIDENTIFIED	Automatic path-protection switch caused by LOP
UNIDENTIFIED	Automatic path-protection switch caused by PDI
UNIDENTIFIED	Automatic path-protection switch caused by SDBER
UNIDENTIFIED	Automatic path-protection switch caused by SFBER
UNIDENTIFIED	Automatic path-protection switch caused by UNEQ
UNIDENTIFIED	Cold restart
UNIDENTIFIED	Exercise ring
UNIDENTIFIED	Exercising ring successfully
UNIDENTIFIED	Far-end exercise ring
UNIDENTIFIED	Exercise request on ring failed
UNIDENTIFIED	Exercise span
UNIDENTIFIED	Exercising span successfully
UNIDENTIFIED	Far-end exercise span
UNIDENTIFIED	Exercise request on span failed
UNIDENTIFIED	Force switch request on facility/equipment
UNIDENTIFIED	Force switch request on ring
UNIDENTIFIED	Force switch request on span
UNIDENTIFIED	Far-end working facility forced to switch to protection unit—ring
UNIDENTIFIED	Far-end working facility forced to switch to protection unit—span
UNIDENTIFIED	Bidirectional full pass-through is active
UNIDENTIFIED	Unidirectional full pass-through is active
EQPT	Inhibit switch to protect request on equipment
EQPT	Inhibit switch to working request on equipment
UNIDENTIFIED	K bytes pass-through is active
UNIDENTIFIED	Far-end lockout of protection—all spans
UNIDENTIFIED	Far-end lockout of protection—ring
UNIDENTIFIED	Far-end lockout of working—ring
UNIDENTIFIED	Far-end lockout of working—span
UNIDENTIFIED	Far-end lockout of protection—span
UNIDENTIFIED	Lockout of protection
UNIDENTIFIED	Lockout of protection—ring
UNIDENTIFIED	Lockout of protection—span
UNIDENTIFIED	Lockout of working—ring
UNIDENTIFIED	Lockout of working—span

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Lockout switch request on facility/equipment
UNIDENTIFIED	Lockout switch request on ring
UNIDENTIFIED	Manual reset
UNIDENTIFIED	Far-end manual switch of working facility to protection unit—ring
UNIDENTIFIED	Far-end manual switch of working facility to protection unit—span
UNIDENTIFIED	Manual switch to fifth reference
UNIDENTIFIED	Manual switch to fourth reference
UNIDENTIFIED	Manual switch to internal clock
UNIDENTIFIED	Manual switch to primary reference
UNIDENTIFIED	Manual switch to second reference
UNIDENTIFIED	Manual switch to sixth reference
UNIDENTIFIED	Manual switch to third reference
UNIDENTIFIED	Manual switch request on facility/equipment
UNIDENTIFIED	Manual switch request on ring
UNIDENTIFIED	Manual switch request on span
UNIDENTIFIED	Powerfail restart
UNIDENTIFIED	Ring is segmented
UNIDENTIFIED	Ring switch is active on the east side
UNIDENTIFIED	Ring switch is active on the west side
UNIDENTIFIED	Span switch is active on the east side
UNIDENTIFIED	Span switch is active on the west side
UNIDENTIFIED	Ring is squelching traffic
UNIDENTIFIED	Squelching path
UNIDENTIFIED	Software download in progress
UNIDENTIFIED	Switch to fifth reference
UNIDENTIFIED	Switch to fourth reference
UNIDENTIFIED	Switch to primary reference
UNIDENTIFIED	Switch to second reference
UNIDENTIFIED	Switch to sixth reference
UNIDENTIFIED	Switch to third reference
ENV	System reboot
UNIDENTIFIED	Extra traffic preempted
UNIDENTIFIED	Switched back to working unit
UNIDENTIFIED	Switched to protection unit
UNIDENTIFIED	Warm restart

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Wait to restore
UNIDENTIFIED	Ring is in wait-to-restore state
UNIDENTIFIED	Span is in wait-to-restore state
UNIDENTIFIED	Bridge and roll has occurred
UNIDENTIFIED	Bridge and roll is pending a valid signal
UNIDENTIFIED	Admin logout of user
UNIDENTIFIED	Admin lockout of user
UNIDENTIFIED	Admin lockout clear
UNIDENTIFIED	Automatic logout of idle user
UNIDENTIFIED	Login of user
SECURITY_VIOLATION	Invalid login—locked out
SECURITY_VIOLATION	Invalid login—already logged on
SECURITY_VIOLATION	Invalid login—password
SECURITY_VIOLATION	Invalid login—username
UNIDENTIFIED	Logout of user
UNIDENTIFIED	User locked out
UNIDENTIFIED	Open I/O slot(s)
UNIDENTIFIED	Improper matrix equipment inserted
EQPT	Equipment fails to boot
UNIDENTIFIED	No description
UNIDENTIFIED	Firewall has been disabled
UNIDENTIFIED	Loss of clock from mate TSC
UNIDENTIFIED	Alarms suppressed for maintenance
UNIDENTIFIED	SNTP host failure
UNIDENTIFIED	Disable inactive user
UNIDENTIFIED	Disable inactive clear
UNIDENTIFIED	Suspend user
UNIDENTIFIED	Duplicate serial number detected on a pluggable entity
EQPT	EQPT problem on carrier or PIM
EQPT	EQPT problem on PIM or PPM
UNIDENTIFIED	Suspend user clear
UNIDENTIFIED	Forced switch to primary reference
UNIDENTIFIED	Forced switch to secondary reference
UNIDENTIFIED	Forced switch to third reference
UNIDENTIFIED	Forced switch to internal clock
PLM	GFP user payload mismatch

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	GFP fibre channel distance extension mismatch
UNIDENTIFIED	GFP fibre channel distance extension buffer starvation
BER_SF	GFP client signal fail detected
LOF	GFP loss of frame delineation
UNIDENTIFIED	GFP extension header mismatch
UNIDENTIFIED	Carrier loss on the LAN
UNIDENTIFIED	Encapsulation type mismatch
UNIDENTIFIED	Transport layer failure
PLM	Payload label mismatch—path
DCC_FAILURE	Line DCC termination failure
UNIDENTIFIED	K byte channel failure
UNIDENTIFIED	BLSR software version mismatch
SECURITY_VIOLATION	Security intrusion attempt detected—see audit log
UNIDENTIFIED	IP address already in use within the same DCC area
UNIDENTIFIED	Node name already in use within the same DCC area
UNIDENTIFIED	Free memory on card near zero
UNIDENTIFIED	Free memory on card very low
ENV	NE power failure at connector
UNIDENTIFIED	Standby database out of sync
UNIDENTIFIED	Database backup failed
UNIDENTIFIED	Database restore failed
UNIDENTIFIED	ISIS adjacency failure
EQPT	Equipment high laser bias
EQPT	Equipment high Rx power
EQPT	Equipment high Tx power
EQPT	Equipment low laser bias
EQPT	Equipment low Rx power
EQPT	Equipment low Tx power
UNIDENTIFIED	Provisioning mismatch
UNIDENTIFIED	Illegal route addition to the network
UNIDENTIFIED	Session time limit expired
UNIDENTIFIED	User password change required
SECURITY_VIOLATION	User authentication rejected
UNIDENTIFIED	APS invalid mode
UNIDENTIFIED	Far-end manual switch back to working—span
UNIDENTIFIED	Far-end forced switch back to working—span

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	Alarms suppressed by user command
UNIDENTIFIED	Ring is squelching STS traffic
UNIDENTIFIED	REPT^DBCHG messages inhibited
TIM	Regenerator section trace identifier mismatch
MS-AIS	AIS—multiplex section—alarm indication signal
RAI	RFI—multiplex section—remote failure/alarm indication
BER_SF	Multiplex section—excessive BER
BER_SD	Multiplex section—signal degrade
DCC_FAILURE	Multiplex section DCC termination failure
UNIDENTIFIED	G811—primary reference clock traceable
UNIDENTIFIED	G812T—transit node clock traceable
UNIDENTIFIED	G812L—local node clock traceable
UNIDENTIFIED	G813—synchronous equipment timing source traceable
UNIDENTIFIED	MS-SPRing out of sync
AU-AIS	AIS—administration unit—alarm indication signal
LOP	LOP—administration unit—loss of pointer
BER_SF	High-order path—excessive BER
BER_SD	High-order path—signal degrade
PLM	SLMF—PLM high order—path label mismatch
TIM	TIM high order—trace identifier mismatch failure
RAI	RFI—high order—remote failure/alarm indication
UNEQ	SLMF—unequipped high order—path unequipped
UNIDENTIFIED	Automatic SNCP switch caused by AIS
UNIDENTIFIED	Automatic SNCP switch caused by LOP
UNIDENTIFIED	Automatic SNCP switch caused by UNEQ
UNIDENTIFIED	Automatic SNCP switch caused by SFBER
UNIDENTIFIED	Automatic SNCP switch caused by SDBER
UNIDENTIFIED	Failure to switch to protection—high-order path
UNIDENTIFIED	MSSP multinode table update completed
UNIDENTIFIED	Bipolar violation
UNIDENTIFIED	High-order path—payload defect indication
UNIDENTIFIED	MSSP software version mismatch
DCC_FAILURE	Regenerator section—DCC termination failure
EQPT	Battery failure
UNIDENTIFIED	Extreme high voltage
UNIDENTIFIED	Extreme low voltage

Table A-28 CTM GateWay/CORBA Mapping for the ONS 15600 SDH (continued)

TMF Mapping	NE Native Probable Cause
UNIDENTIFIED	High voltage
UNIDENTIFIED	Low voltage
UNIDENTIFIED	Voltage reading mismatch between SC cards
EQPT	Wavelength out of lock
EQPT	Automatic laser shutdown
UNIDENTIFIED	Ring is squelching high-order traffic
EQPT	Non-Cisco PPM inserted
EQPT	Unqualified PPM inserted

APPENDIX B

Server Administration and Configuration

This appendix includes the following information on the administration and configuration of CTM GateWay/CORBA:

- B.1 Creating an OSS Client Profile for CTM GateWay/CORBA, page B-1
- B.2 Deleting an OSS Client Profile for CTM GateWay/CORBA, page B-2
- B.3 Viewing Currently Logged In CTM GateWay/CORBA OSS Users, page B-2
- B.4 Logging Out CTM GateWay/CORBA OSS Users, page B-3
- B.5 Using Encryption Between the OSS Client and CTM GateWay/CORBA, page B-3
- B.6 Using Multiple Naming Servers, page B-4
- B.7 Naming Conventions for Published CTM GateWay/CORBA Objects, page B-5
- B.8 Location of the Naming Service IOR File, page B-6
- B.9 Useful Debugging Utilities for Resolving Naming Service-Related Issues, page B-6
- B.10 Configuring CTM GateWay/CORBA, page B-6
- B.11 Using the CLI to Start and Stop CTM GateWay/CORBA, page B-8
- B.12 Configuring Secure Socket Layer for CTM GateWay/CORBA, page B-9
- B.13 Installation Program, page B-12
- B.14 CTM R9.1-to-CTM R9.2 Migration, page B-12

B.1 Creating an OSS Client Profile for CTM GateWay/CORBA

The CORBA gateway authenticates the OSS against a previously created user profile before allowing access to CTM. You can create OSS client profiles for CTM GateWay/CORBA sessions. Each OSS profile defines CTM GateWay/CORBA parameters, such as the OSS profile name, password, and IP address.

OSS client profiles are stored in the CTM GateWay/CORBA Users table.

Step 1 Log into the CTM client with administrator privileges.

Step 2 In the Domain Explorer window, choose **Administration > GW/CORBA Users**.

Step 3 Choose **Edit > Add** (or click the **Create a New User** tool).

Step 4 In the Add GW/CORBA User window, enter the following OSS client information:

- OSS Profile Name—Name of the OSS profile.
- Password—Password that the OSS client uses to log into the CTM server. The password must contain at least one special character, at least two alphabetic characters (A-Z, a-z), and at least one numeric character (0-9). Apostrophes (‘) are not accepted.
- Confirm Password—Re-enter the password to confirm it.

Step 5 Click **OK** to confirm the information. Changes take effect immediately. The GW/CORBA Users table receives a refresh event. If automatic refresh is enabled, the new OSS client profile appears as a new row in the table. If automatic refresh is disabled, click the **Refresh Data** tool to see the new OSS client profile in the table.

Step 6 In the Control Panel window, choose **Administration > GW/CORBA Users**. The GW/CORBA Users wizard displays a profile for each OSS client that uses a CTM GateWay/CORBA service.

B.2 Deleting an OSS Client Profile for CTM GateWay/CORBA

Step 1 Log into the CTM client with administrator privileges.

Step 2 In the Domain Explorer window, choose **Administration > GW/CORBA Users**.

Step 3 The GW/CORBA Users table displays a list of available OSS users. Select the OSS user to delete.

Step 4 Choose **Edit > Delete** (or click the **Delete a User** tool) to delete the OSS profile from the CTM database.

Step 5 Click **OK** to confirm the deletion. The OSS client profile name is deleted from the GW/CORBA Users table.

Note If the OSS is connected to CTM when the profile is being deleted, CTM does not terminate the OSS session.

B.3 Viewing Currently Logged In CTM GateWay/CORBA OSS Users

Step 1 Log into the CTM client with administrator privileges.

Step 2 In the Domain Explorer window, choose **Administration > GW/CORBA Users**

Step 3 The GW/CORBA Users table displays a list of available OSS users. Click the **Show Logged in GW CORBA Users** tool.

Step 4 In the Active GW/CORBA Users table, a list of currently logged-in users is displayed, including the OSS profile name, IP address to which the user is logged in, and the login time.

B.4 Logging Out CTM GateWay/CORBA OSS Users

Step 1 Log into the CTM client with administrator privileges.

Step 2 In the Domain Explorer window, choose **Administration > GW/CORBA Users**.

Step 3 The GW/CORBA Users table displays a list of available OSS users. Click the **Show Logged in GW CORBA Users** tool.

Step 4 In the Active GW/CORBA Users table, a list of currently logged-in users is displayed, including the OSS profile name, IP address to which the user is logged in, and the login time. Select the user to log off.

Step 5 Click the **Log Out GW CORBA User** tool.

Step 6 The user session is cleared from the Active GW/CORBA Users table. You will notice the loss of session during the next ping cycle or when you try to perform an operation on another manager. Some examples of user operations are:

- The user is connected and performs a query operation on the EMS. The OSS user starts to query the EMS by getting a fresh object reference from the manager through an emsSession query. Because the session has been cleared by the CTM GateWay/CORBA service, the OSS user receives an exception and notices the loss of session.
- The CTM client forcefully logs out the user. The OSS user does not immediately notice the loss of session when the CTM client forces a logout. To immediately log out the user, the CTM GateWay/CORBA service makes a call to the NMS session interface, which forces the OSS client applications to modify their shutdown application. This is not the preferred method.
- The CTM GateWay/CORBA service clears the user session information from its internal memory and database.

B.5 Using Encryption Between the OSS Client and CTM GateWay/CORBA

CTM uses improved encryption of usernames and passwords for network security. You can set the Control Panel to send encrypted usernames and passwords to CTM GateWay/CORBA:

Step 1 Log into the CTM client with administrator privileges.

Step 2 In the Domain Explorer window, choose **Administration > Control Panel**.

Step 3 Click the **GateWay/CORBA Service** tab for the GateWay/CORBA Service property sheet.

Step 4 Click the **Global** tab and check the **Enable Encryption for Username and Password check** box.

Step 5 Click **Save**; then, click **Yes** in the confirmation dialog box. Changes take effect immediately.

If the OSS clients enable the encryption feature, they must provide implementation for RSA-based encryption by retrieving the RSA public key or the public key pair from CTM GateWay/CORBA and by using cryptographic libraries.

- To obtain the RSA public key from CTM, use the emsSessionFactory::EmsSessionFactory_I:: getEmsPublicKey API. See 3.5.2 getEmsPublicKey, page 3-53.
- To obtain the RSA public key pair from CTM, use the emsSessionFactory::EmsSessionFactory_I:: getEmsPublicKeyPair API. See 3.5.3 getEmsPublicKeyPair, page 3-54.

CTM uses a 512-bit (64-byte) key size and returns the string representation of the RSA public key or public key pair, encoded in the Base64 encoding scheme. OSS clients should use Base64 decoders to decode the public key and get the byte[] of the public key from the decoded public key string. The byte[] corresponding to the public key represents the key in its primary encoded format (X.509 SubjectPublicKeyInfo). Using this byte[] and cryptographic libraries, the RSA public key can be created.

One example of the security provider is Bouncy Castle Provider.

Use the public key to encrypt the username and password. Before passing the encrypted username and password to CTM for login, OSS clients should encode the encrypted username and password by using Base64 encoders to obtain the string equivalent of the encrypted data.

Note Use cryptographic libraries implementing RSA public key encryption supporting the "PKCS #1 v2.0 EME-PKCS1-v1_5 (PKCS #1 v1.5 block type 2), PKCS1Padding" encoding scheme. CTM does not provide these cryptographic libraries.

B.6 Using Multiple Naming Servers

CTM can register with multiple naming servers. You must add the following parameters to the *CTM-server-installation-directory*/cfg/corbagw.properties file:

- corbagw.namingservice.ServerList=ctmc4-u80,ctm7-u60
- corbagw.namingservice.RootIORLoc=/namingroot.ior

Complete the following steps to allow CTM to use multiple naming servers:

Step 1 In the Domain Explorer window, choose **Administration > Control Panel**.

Step 2 Click **GateWay/CORBA Service** to open the GateWay/CORBA Service pane.

Step 3 In the **Global** tab > **GateWay/CORBA Configuration** area, specify the following parameters:

- Name Service Server List—Lists all the hosts on which the naming service is running. The hosts should be reachable from the CTM server host, and the HTTP server must be running on all naming service hosts. Enter **ctmc4-u80, ctm7-u60**.

  Note In addition to these naming service hosts, CTM registers itself with the local naming service. The local naming service port is 14005 and is bundled with CTM.

- Name Service Root IOR—Defines the location and name of the file that contains the naming service root Interoperable Object Reference (IOR). The IOR file must be accessible through the following HTTP call: http://*name-server-IP-address*:80/namingroot.ior. Enter **/namingroot.ior**.

Step 4 Restart the CTM GateWay/CORBA service.

B.7 Naming Conventions for Published CTM GateWay/CORBA Objects

CTM GateWay/CORBA publishes two top-level objects: EMSSessionFactory and NotificationChannel. CTM creates these objects and registers them with the CORBA name server.

CTM GateWay/CORBA creates naming contexts under the root as shown in Figure B-1. The last context in the tree must have a different name. To change this value in the CTM client GUI:

Step 1 Log into the CTM client with the appropriate CTM user access profile.

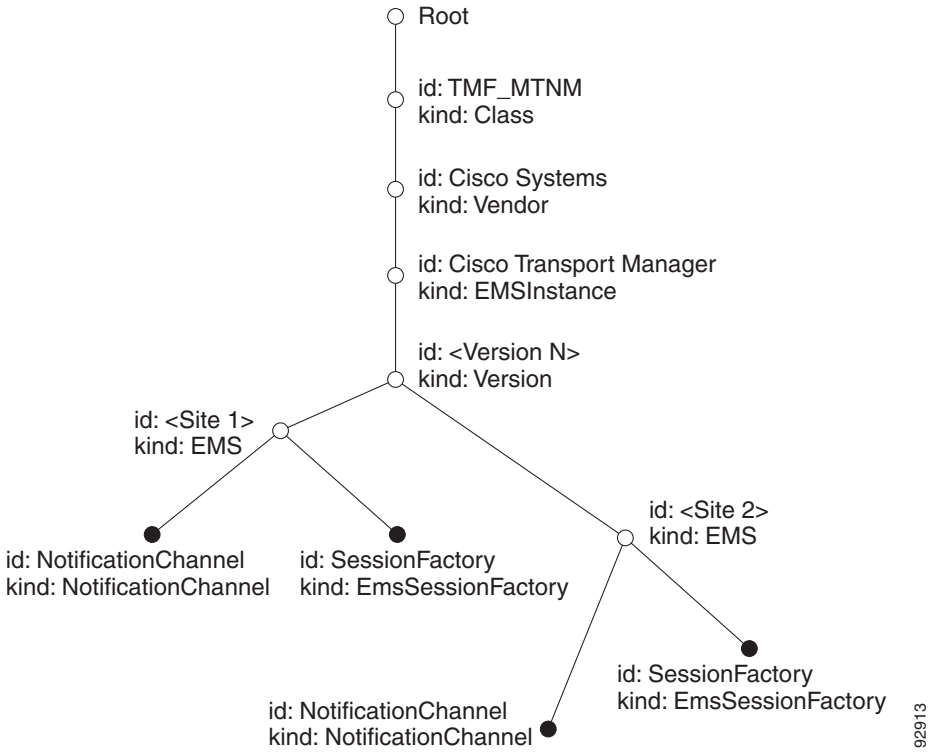
Step 2 In the Domain Explorer window, click the **CTM Domain** node.

Step 3 In the Management Domain Properties sheet, click the **Identification** tab.

Step 4 In the EMS Domain section, look for **EMS ID**. The value of this field should be used as the "id" field for context, where "kind" equals "EMS." The default value is *CTM*. By using different names, you can install multiple instances of CTM and use a centralized naming server and repository.

The following figure shows the naming scheme for CTM GateWay/CORBA objects.

Figure B-1 Naming Scheme for CTM GateWay/CORBA Objects



B.8 Location of the Naming Service IOR File

The naming service IOR is located at:

/opt/CiscoTransportManagerServer/openfusion/domains/OpenFusion/localhost/NameService/NameSingleton/NameSingleton.ior

B.9 Useful Debugging Utilities for Resolving Naming Service-Related Issues

The following are samples of CTM commands (bundled utility programs) for debugging naming service connectivity issues.

Obtain the list of registered objects in the OpenFusion naming service:

```
setenv PATH /opt/CiscoTransportManagerServer/openfusion/bin:$PATH
setenv NS_IOR_LOCATION
file:///opt/CiscoTransportManagerServer/openfusion/domains/OpenFusion/localhost/NameServic
e/NameSingleton/NameSingleton.ior
nsMgrTool -l
```

Decode an IOR file:

```
setenv PATH /opt/CiscoTransportManagerServer/openfusion/bin:$PATH
dior -f <IOR file name>, or
dior -i <IOR string>
```

Check if the naming service is running:

```
setenv PATH /opt/CiscoTransportManagerServer/openfusion/bin:$PATH
server -status NameService
```

Note The /opt/CiscoTransportManagerServer/openfusion/bin directory contains nsMgrTool, dior, and server utility tools.

B.10 Configuring CTM GateWay/CORBA

You can configure the following CTM GateWay/CORBA properties in the CTM client Control Panel.

Step 1 From the Domain Explorer, choose **Administration > Control Panel**.

Step 2 In the Control Panel, click **GateWay/CORBA Service**. Configure the following properties:

Note If CTM GateWay/CORBA is running, changes to the config file do not take effect dynamically. You must restart CTM GateWay/CORBA for the changes to take effect.

- Enable Encryption for username and password:

  This property defines whether to encrypt the username and password used for the CTM GateWay/CORBA client.

- Heartbeat for Notification Channel (min): 0

  This property is the rate at which the notification service is checked. A zero entry means not to check the notification service.

- Enter the maximum number of simultaneous sessions: 4

  This property is the number of CTM GateWay/CORBA sessions that can be active at the same time. The range is from 4 to 25 sessions.

- Enter the maximum events per consumer: 10000

  CTM GateWay/CORBA uses this property to set the MaxEventsPerConsumer administrative QoS parameter of the notification channel. The notification server uses this property to bound the maximum number of events in a given channel that are allowed to queue at any given time. The default value is 0, where the notification server does not impose a limit on the maximum number of events that can be queued. If no limits are imposed on the queue, the notification server might run out of memory if a client behaves incorrectly. The server must keep all events in memory until they are consumed by all registered consumers.

Caution Any change to this value should be made with extreme caution. If you set the value too low, the NMS will not receive all notifications. If you set the value too high, the CTM notification server will run out of memory. The current value is set to handle alarm bursts of 10,000 events per minute.

- Enter the notification service name: NotificationService

  This property defines the service name that the resolve_initial_reference function uses to get a reference to the notification service. The CTM GateWay/CORBA installation installs the notification service automatically. To use your own notification service, modify this parameter.

Tip You do not need to change this parameter if you plan to use the notification service that is bundled with CTM GateWay/CORBA.

- Enter the notification service naming context: services/NotifyChannelFactory

  NamingContext defines the naming context of NotificationService. This property is used when resolve_initial_reference fails to resolve NotificationService. CTM GateWay/CORBA contacts the naming service to resolve the name context defined in this property. The value of this property must match the value published by your notification server.

Tip You do not need to change this parameter if you plan to use the notification service that is bundled with CTM GateWay/CORBA.

- Enter the notification service factory IOR filename: file:/opt/CiscoTransportManagerServer/openfusion/domains/OpenFusion/localhost/NotificationService/NotificationSingleton/NotificationSingleton.ior

  The FactoryIORFile property defines the path to a text file that contains the IOR of NotificationService. This property is used only after resolve_initial_reference and naming service fail. CTM GateWay/CORBA opens the file as defined by the URL format in this property and attempts to retrieve the IOR from this file. This parameter lets you run your notification service on a different host to improve performance.

Tip You do not need to change this parameter if you plan to use the notification service that is bundled with CTM GateWay/CORBA.

- Enter the notification service listening port number: 0

  This property is used to set the port that the notification service uses to listen for incoming requests. The port number is set in the IOR for the notification service. The use IOR and use IOR endpoint properties are set correctly. The default port number is zero, which signifies the port number allocated by the operating system.

- Enter the session port number: 0

  This property configures the EMSSessionFactory port. If this property is set to zero, the operating system allocates the session port number.

- Enter the name service server list:

  This property defines where the name servers are running. This property accepts a comma-separated list of hostnames.

- Enter the name service root IOR:

  This property defines the path used to find the naming service IOR on each host defined in ServerList. The complete path is constructed as http://*item-of-ServerListRootIORLoc*

- Error level: Minor

  This property defines the error level of messages to log.

  You can configure this CTM GateWay/CORBA property by modifying a configuration file in *CTM-server-installation-directory*/cfg/corbagw.properties.

- corbagw.CTP.getLayeredParameters=false

  By default, this property is not enabled. If the NMS requires CTP-related transmission parameters to be included as part of an object reporting TerminationPoint_T structure, this property must be set to true. However, the ManagedElementMgr_I.getTP interface always returns transmission parameters as part of the TerminationPoint_T structure and is independent of this property setting.

B.11 Using the CLI to Start and Stop CTM GateWay/CORBA

CTM can manage the CTM GateWay/CORBA service from the command line:

- To start a CTM GateWay/CORBA service, run the /opt/CiscoTransportManagerServer/bin/gwcorba-start script from the command line.
- To stop a CTM GateWay/CORBA service, run the /opt/CiscoTransportManagerServer/bin/gwcorba-stop script from the command line.

Only CTM users with administrative privileges can run these scripts. If the CTM GateWay/CORBA service is already running and you attempt to run the gw-start script, the script exits with the message "GWCORBA already running." If the CTM GateWay/CORBA service is stopped and you attempt to run the gw-stop script, the script exits with the message "GWCORBA not running."

You must have a CTM username and password with a SysAdmin or SuperUser profile to start or stop the scripts.

B.12 Configuring Secure Socket Layer for CTM GateWay/CORBA

To ensure network security, CORBA calls can be made over Secure Socket Layer (SSL).

The current JacORB implementation is precompiled with JacORB security libraries. To configure SSL for CTM GateWay/CORBA, you must set up a keystore and configure the properties in the client-side jacorb.properties file.

The client must enforce SSL by modifying the jacorb.properties file. The server-side keystore is generated using the JSSE keystore. CTM bundles a default keystore and a certificate for the CTM GateWay/CORBA service.

As explained in the following sections, you must generate the server-side certificate and add it to the client-side keystore; then generate and add the client-side certificate to the server-side keystore.

B.12.1 Generating the Server-Side Certificate

Step 1 Enter the **keytool** command to generate a keystore and a key:

```
keytool -genkey -alias gwcorba_service -validity 25000 -keystore gwcorba_service_ks
-storepass gwcorba_service_ks_pass -keypass gwcorba_service_ks_pass

What is your first and last name?
  [Unknown]:  gateway corba server
What is the name of your organizational unit?
  [Unknown]:
What is the name of your organization?
  [Unknown]:  cisco
What is the name of your City or Locality?
  [Unknown]:
What is the name of your State or Province?
  [Unknown]:
What is the two-letter country code for this unit?
  [Unknown]:
Is <CN=gateway corba server, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown>
correct?
  [no]:  y
```

Step 2 Verify that the generated keystore and key have the following attributes:

```
Keystore name: gwcorba_service_ks
Alias: gwcorba_service
Keystore password: gwcorba_service_ks_pass
Key password: gwcorba_service_ks_pass
Validity: 25000 days
```

Step 3 Enter the following command to generate a server-side certificate that will be issued to the client:

```
keytool -export -keystore gwcorba_service_ks -alias gwcorba_service -storepass
gwcorba_service_ks_pass -file gwcorba_service_cert

Certificate stored in file <gwcorba_service_cert>
```

Step 4 Verify that the certificate is stored in the gwcorba_service_cert file. The server-side certificate and keystore are present in the /opt/CiscoTransportManagerServer/cfg directory.

B.12.2 Generating the Client-Side Certificate

Note the following conventions:

- ascii_client_ks—Denotes a client-side keystore.
- ascii_client_cert—Denotes a client-side certificate.
- gwcorba_service_ks—Denotes a server-side keystore.
- gwcorba_service_cert—Denotes a server-side certificate.

Step 1 Enter the **keytool** command to generate a keystore:

```
keytool -genkey -alias ascii_client -validity 25000 -keystore ascii_client_ks -storepass
ascii_client_ks_pass -keypass ascii_client_ks_pass

What is your first and last name?
  [Unknown]:  ascii client
What is the name of your organizational unit?
  [Unknown]:
What is the name of your organization?
  [Unknown]:  cisco
What is the name of your City or Locality?
  [Unknown]:
What is the name of your State or Province?
  [Unknown]:
What is the two-letter country code for this unit?
  [Unknown]:
Is CN=ascii client, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown correct?
[no]:  y
```

Step 2 Verify that the generated keystore and key have the following attributes:

```
Keystore name: ascii_client_ks
Alias: ascii_client
Keystore password: ascii_client_ks_pass
Key password: ascii_client_ks_pass
Validity: 25000 days
```

Step 3 Enter the following command to generate a client-side certificate that will be issued to the server:

```
keytool -export -keystore ascii_client_ks -alias ascii_client -storepass
ascii_client_ks_pass -file ascii_client_cert

Certificate stored in file <ascii_client_cert>
```

Step 4 Verify that the certificate is stored in the ascii_client_cert file.

B.12.3 Adding the Client-Side Certificate to the Server-Side Keystore

Step 1 Enter the following command to add the client-side certificate to the server-side keystore. (Use FTP or a similar tool to deliver the ascii_client_cert file to the server. The server-side keystore is located in the /opt/CiscoTransportManagerServer/cfg directory on the server.)

```
keytool -import -keystore gwcorba_service_ks -alias ascii_client -storepass
gwcorba_service_ks_pass -file ascii_client_cert
```

The command output resembles the following example:

```
Owner: CN=ascii client, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown
Issuer: CN=ascii client, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown
Serial number: serial-number
Valid from: Sat Mar 18 17:18:44 GMT+05:30 2008 until: Fri Jul 23 10:50:28 GMT+05:30 2038
Certificate fingerprints:
    MD5: 42:53:98:7C:BA:CB:28:39:50:50:9F:E4:56:F2:43:FF
    SHA1: F5:1D:B9:BB:1D:66:C2:A1:32:BE:47:0C:85:47:17:16:A2:69:17:4C
Trust this certificate? [no]: y
Certificate was added to keystore
```

Step 2 Verify that the certificate issued by the client was added to the server keystore.

B.12.4 Adding the Server-Side Certificate to the Client-Side Keystore

Step 1 Enter the following command to add the server-side certificate to the client-side keystore. (Use FTP or a similar tool to deliver the previously generated gwcorba_service_cert file from the server. The server-side certificate is located in the /opt/CiscoTransportManagerServer/cfg directory on the server.)

```
keytool -import -keystore ascii_client_ks -alias gwcorba_service -storepass
ascii_client_ks_pass -file gwcorba_service_cert
```

The command output resembles the following example:

```
Owner: CN=gateway corba server, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown
Issuer: CN=gateway corba server, OU=Unknown, O=cisco, L=Unknown, ST=Unknown, C=Unknown
Serial number: serial-number
Valid from: Sat Mar 18 17:21:24 GMT+05:30 2008 until: Fri Jul 23 10:53:08 GMT+05:30 2038
Certificate fingerprints:
    MD5: 5C:41:39:AD:D0:F8:63:5D:81:8D:47:A0:33:02:8E:7D
    SHA1: 38:CD:C8:57:F7:15:22:DC:1A:6E:99:CD:13:A1:9A:67:90:2C:65:C2
Trust this certificate? [no]: y
Certificate was added to keystore
```

Step 2 Verify that the certificate issued by the server was added to the client keystore.

B.12.5 Configuring the Client-Side Properties

To enforce SSL, complete the following configuration on the client-side jacorb.properties file.

Note

- To enforce SSL, the supported and required options must be set to 60.
- No changes are required to the server-side jacorb.properties file, because it has already been changed.

```
jacorb.security.support_ssl=on

# IIOP/SSL parameters (numbers are decimal values):
# EstablishTrustInClient = 40
# EstablishTrustInTarget = 20
# mutual authentication = 60
```

```
jacorb.security.ssl.client.supported_options=60
jacorb.security.ssl.client.required_options=60

jacorb.ssl.socket_factory=org.jacorb.security.ssl.sun_jsse.SSLSocketFactory

jacorb.security.keystore_password= ascii_client_ks_pass
jacorb.security.keystore= ascii_client_ks

# Read trusted certificates from the keystore
jacorb.security.jsse.trustees_from_ks=on
```

B.13 Installation Program

The CTM installation program installs the CTM GateWay/CORBA component, which includes OpenFusion 4.2.3 (JacORB, Notification Service, and Name Service) from Prism Technologies.

Interface Definition Language (IDL) files are installed under the /opt/CiscoTransportManagerServer/idl directory. See the *Cisco Transport Manager Release 9.2 Installation Guide* for more information.

B.14 CTM R9.1-to-CTM R9.2 Migration

All procedures related to the migration from CTM R9.1 to CTM R9.2 are performed during the installation phase; no additional operations are required.

APPENDIX C

OSS Use Cases and Client Development

This appendix lists interceptors and use cases to resynchronize the NMS from the EMS. This appendix includes the following sections:

- C.1 Use Cases to Resynchronize the NMS from the EMS, page C-1
- C.2 Developing a CTM GateWay/CORBA Client, page C-4

C.1 Use Cases to Resynchronize the NMS from the EMS

Table C-1, Table C-2, and Table C-3 list use cases to resynchronize the NMS from the EMS.

Table C-1 Use Cases to Resynchronize the NMS from the EMS

Information	Detail
Name	The NMS retrieves CTM information.
Summary	The NMS retrieves all EMS and ME alarms and information for existing multilayer subnetworks, MEs, PTPs, CTPs, SNCs, and protection groups from CTM GateWay/CORBA interfaces.
Actor(s)	NMS.
Preconditions	The NMS locates the EmsSessionFactory object and obtains references to EMSMgr_I, MultiLayerSubnetworkMgr_I, ManagedElementMgr_I, EquipmentInventoryMgr_I, MaintenanceMgr_I, ProtectionMgr_I, and PerformanceManagementMgr_I interfaces.
Begins When	The NMS sends a request to retrieve all of the CTM information.

Table C-1 Use Cases to Resynchronize the NMS from the EMS (continued)

Information	Detail
Description	The NMS does the following to request information from the CTM GateWay/CORBA interface: 1. Uses the emsMgr::EMSMgr_I:: getAllEMSSystemActiveAlarms interface to request all EMS alarms. 2. Uses the emsMgr::EMSMgr_I:: getAllTopLevelSubnetworks interface to request all multilayer subnetworks in CTM. CTM returns a list of MultiLayerSubnetwork_T objects with detailed information about a multilayer subnetwork. 3. Uses the emsMgr::EMSMgr_I:: getAllTopLevelTopologicalLinks interface to request all topological links across multiple multilayer subnetworks. CTM returns a list of TopologicalLink_T objects with detailed information about a topological link. 4. For each MultiLayerSubnetwork_T object, CTM uses the multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: getAllManagedElements interface to request all MEs that belong to that multilayer subnetwork. CTM returns a list of ManagedElement_T objects with detailed information about an ME. For each ManagedElement object, executes the use case "NMS Resynchronizes Information Specific to a Managed Element." 5. Uses the multiLayerSubnetwork::MultiLayerSubnetworkMgr_I:: getAllVLANs interface to request all VLANs associated with the multilayer subnetwork.
Ends When	The NMS retrieves all CTM information.
Exceptions	Refer to the exceptions thrown by the individual interface method.
Post Conditions	• The NMS synchronizes with CTM. • The NMS registers with CTM to retrieve notifications related to changes in the managed object on CTM, new alarms, and TCAs.
Traceability	—

Table C-2 NMS Resynchronizes Information Specific to a Managed Element

Information	Detail
Name	The NMS retrieves CTM information specific to the ME.
Summary	The NMS retrieves all PTP, CTP, SNC, and protection group information for the specific ME.
Actor(s)	NMS.
Preconditions	The NMS obtains references to ManagedElementMgr_I, EquipmentInventoryMgr_I, MaintenanceMgr_I, ProtectionMgr_I, and PerformanceManagementMgr_I interfaces. The NMS also obtains the ME name.
Begins When	The NMS sends a request to retrieve all information about an ME. The NMS identifies the name of the ME for which to retrieve information.

Table C-2 NMS Resynchronizes Information Specific to a Managed Element (continued)

Information	Detail
Description	The NMS does the following to request information from the CTM GateWay/CORBA interface: 1. Uses the managedElementManager::ManagedElementManager_I:: getAllActiveAlarms interface to request all current alarms on the ME. 2. Uses the equipment::EquipmentInventoryMgr_I:: getAllEquipment interface to request all existing equipment on the ME. CTM returns a list of EquipmentHolder_T objects with detailed information about equipment holders and equipment on the ME. 3. For the equipment information retrieved in EquipmentorHolder_T object, uses the equipment::EquipmentInventoryMgr_I:: getAllSupportedPTPs interface to request all supported PTPs on the equipment. CTM returns a list of TerminationPoint_T objects with detailed PTP information, such as SDH or SONET port, admin state, service state, line code, frame format, and so on. 4. For each TerminationPoint_T object, executes the "NMS Resynchronizes Information Specific to TerminationPoint" use case. 5. Uses the managedElementManager::ManagedElementManager_I:: getAllSNCs interface to request all SNCs originating, terminating, or passing through the ME. CTM returns a list of SubnetworkConnection_T objects with detailed SNC information, such as layer rate, SNC state, SNC protection state, a-end CTP, z-end CTP, and so on. 6. Uses the protection::ProtectionMgr_I:: getAllProtectionGroups interface to request all protection groups created on the ME. CTM returns a list of ProtectionGroup_T objects with detailed protection group information, such as 1:1, 1+1, or 2F-BLSR; revertive or nonrevertive; and so on.
Ends When	The NMS retrieves all ME-specific information.
Exceptions	Refer to the exceptions thrown by the individual interface method.
Post Conditions	The NMS synchronizes information about an ME with CTM.
Traceability	—

Table C-3 NMS Resynchronizes Information Specific to a Termination Point

Information	Detail
Name	The NMS retrieves CTM information specific to the termination point.
Summary	The NMS retrieves all loopback and threshold information for PTPs and CTPs used by the SNC.
Actor(s)	NMS.
Preconditions	The NMS obtains references to ManagedElementMgr_I, MaintenanceMgr_I, and PerformanceManagementMgr_I interfaces. The NMS also obtains a list of PTPs.
Begins When	The NMS sends a request to retrieve all threshold and loopback information on a PTP and on any in-use CTPs. The NMS identifies the name and type (PTP/CTP) of the TP for which to retrieve information.

Table C-3 NMS Resynchronizes Information Specific to a Termination Point (continued)

Information	Detail
Description	The NMS does the following to request information from the CTM GateWay/CORBA interface: 1. Uses the maintenanceOps::MaintenanceMgr_I:: getActiveMaintenanceOperations interface to request all loopback information on the TP. CTM returns a list of CurrentMaintenanceOperation_T objects with detailed information about the type of loopback set on the PTP, if any. CTM supports two types of loopback, FACILITY_LOOPBACK and TERMINAL_LOOPBACK. If there are no loopbacks set on the TP, the returned list is empty. 2. Uses the performance::PerformanceManagementMgr_I:: getTCATPParameter interface to request the threshold values set for all PM thresholds on the given TP object. CTM returns the TCAParameter_T object with detailed information about all PM threshold name/value pairs for the PTP. 3. For each PTP that the NMS specifies as the TP, uses the managedElementManager::ManagedElementManager_I:: getContainedInUseTPs interface to request all CTPs contained in the PTP participating in an SNC. CTM returns a list of TerminationPoint_T objects with detailed information about the CTPs participating in an SNC and the associated provisioning details, such as IPPM monitor status, J1 path trace status, and so on. 4. For each CTP object, uses the performance::PerformanceManagementMgr_I:: getTCATPParameter interface to request threshold information. CTM returns a list of TCAParameter_T objects with detailed information about the PM threshold name/value pairs for the CTP.
Ends When	The NMS retrieves loopback and threshold information for all PTPs and CTPs.
Exceptions	Refer to the exceptions thrown by the individual interface method.
Post Conditions	The NMS synchronizes information about a TP object with CTM.
Traceability	—

C.2 Developing a CTM GateWay/CORBA Client

CTM GateWay/CORBA implements APIs defined by the TeleManagement Forum's Multi Technology Network Management (MTNM) group. These APIs are defined for communication between an NMS and the EMS. The NMS must develop a client application that uses these APIs. The following sections describe the tools required for developing a client application. Sample code is provided.

C.2.1 Recommended Tools

You can develop the CORBA client on UNIX or PC platforms. Cisco recommends the following development tools:

- Sun Microsystems Java Development Kit (JDK) 1.5 (available at http://java.sun.com/)
- jacORB 2.x

C.2.2 Sample Code in Java

A typical CORBA client application involves the following steps:

Step 1 Initialize the client connection to the object request broker (ORB).

Step 2 Obtain a reference to the naming service.

Step 3 Resolve the reference to EMSSessionFactory. See C.2.2.3 Get Reference to EMSSessionFactory, page C-7.

Step 4 Implement NmsSession_IOperations. See C.2.2.4 Implement NmsSession_IOperations, page C-7.

Step 5 Retrieve EmsSession by supplying the username and password. See C.2.2.5 Log In and Retrieve EmsSession, page C-8.

Step 6 Query EmsSession and obtain a list of managers available for operations. See C.2.2.6 Retrieve List of Managers, page C-9.

Step 7 Invoke the desired method on that manager. See C.2.2.7 getEMS Operation on EMS Manager, page C-9.

If you want your client to receive notifications from CTM GateWay/CORBA, the following additional steps are required:

Step 8 Obtain a reference to an EventChannel object in the notification server in Step 2.

Step 9 Obtain a ConsumerAdmin object reference.

Step 10 Invoke obtain_notification_push_consumer() on the SupplierAdmin object, specifying CosNotifyChannelAdmin::STRUCTURED_EVENT as a parameter. This operation returns a reference to the StructuredProxyPushSupplier.

Step 11 Implement an instance of StructuredPushConsumer (defined by the OMG).

Step 12 Invoke the connect_structured_push_consumer() operation on the StructuredProxyPushSupplier object to connect the StructuredPushConsumer implementation object to the notification channel.

Step 13 Monitor incoming notifications.

The following sections show sample Java code for the steps required to develop a client application.

Note The sample code shown is for illustrative purposes only; the code might not compile as shown. The sample code does not handle all exceptions.

C.2.2.1 Initialize the Client Connection

```
// Import OMG packages
import org.omg.CORBA.IntHolder;
import org.omg.CORBA.ORB;
import org.omg.CORBA.Policy;

// Import naming context packages.
import org.omg.CosNaming.NameComponent;
import org.omg.CosNaming.NamingContextExt;
import org.omg.CosNaming.NamingContextExtHelper;

// Import notify channel and event service packages.
import org.omg.CosEventChannelAdmin.AlreadyConnected;
```

```
import org.omg.CosEventChannelAdmin.TypeError;
import org.omg.CosNotifyChannelAdmin.AdminLimitExceeded;
import org.omg.CosNotifyChannelAdmin.AdminNotFound;
import org.omg.CosNotifyChannelAdmin.ClientType;
import org.omg.CosNotifyChannelAdmin.ConsumerAdmin;
import org.omg.CosNotifyChannelAdmin.EventChannel;
import org.omg.CosNotifyChannelAdmin.EventChannelHolder;
import org.omg.CosNotifyChannelAdmin.ProxySupplier;
import org.omg.CosNotifyChannelAdmin.StructuredProxyPushSupplier;
import org.omg.CosNotifyChannelAdmin.StructuredProxyPushSupplierHelper;
import org.omg.CosNotifyComm.StructuredPushConsumer;
import org.omg.CosNotifyComm.StructuredPushConsumerPOATie;

// Import POA packages
import org.omg.PortableServer.POA;
import org.omg.PortableServer.POAHelper;
import org.omg.PortableServer.POAPackage.ServantAlreadyActive;
import org.omg.PortableServer.POAPackage.WrongPolicy;

// Import TMF packages
import org.tmforum.mtnm.emsSession.EmsSession_I;
import org.tmforum.mtnm.emsSession.EmsSession_IHolder;
import org.tmforum.mtnm.emsSession.EmsSession_IPackage.managerNames_THolder;
import org.tmforum.mtnm.emsSessionFactory.EmsSessionFactory_I;
import org.tmforum.mtnm.emsSessionFactory.EmsSessionFactory_IHelper;
import org.tmforum.mtnm.nmsSession.NmsSession_I;
import org.tmforum.mtnm.nmsSession.NmsSession_IPOATie;
public static void main(String[] args)
{

    try {
        // Optional: set up ORB properties: you can set them here or in the launch file
        for the client
        // Properties sys_properties = System.getProperties();

        // For JacORB
        // sys_properties.put("org.omg.CORBA.ORBClass",
    "org.jacorb.orb.ORB"
        // sys_properties.put("org.omg.CORBA.ORBSingletonClass",
    "org.jacorb.orb.ORBSingleton"

        // For Orbix
        // sys_properties.put("org.omg.CORBA.ORBClass",
    "com.iona.corba.art.artimpl.ORBImpl");
        // sys_properties.put("org.omg.CORBA.ORBSingletonClass",
    "com.iona.corba.art.artimpl.ORBSingleton");

        // For Visibroker
        // sys_properties.put("org.omg.CORBA.ORBClass",
    "com.inprise.vbroker.orb.ORB"
        // sys_properties.put("org.omg.CORBA.ORBSingletonClass",
    "com.inprise.vbroker.orb.ORBSingleton"

        // For other ORBs check the related ORB programming guide


        // Step 1: Initialize the ORB and obtain ROOT POA reference
        // Note: ORB_init will process any -ORB arguments
        // be called before any other argument processing.
        //
        global_orb = ORB.init(args, null);
        org.omg.CORBA.Object root_poa = global_orb.resolve_initial_references("RootPOA");
        POA rpoa = POAHelper.narrow(root_poa);
        POA poa = rpoa.create_POA("myPolicy",null, new Policy[0]);
```

```
        poa.the_POAManager().activate();
    }
    catch (SystemException ex) {
        // Exception handling
    }
}
```

C.2.2.2 Get Reference to the Naming Service

```
// Get Nameservice reference
NamingContext nsRootContext = null;
    try {
    // Step 2: Get reference to the name service
    // Option 1: Resolve initial reference (RIR)
    // org.omg.CORBA.Object obj = global_orb.resolve_initial_references("NameService");
    // or

     // Option 2: corbaloc URL
     String objRef = "corbaloc:iiop:gatewayserver.cisco.com:14005/NameServiceGWC";
    org.omg.CORBA.Object obj = global_orb.string_to_object(objRef);
    /*  NOTE: Please replace "gatewayserver.cisco.com" with the name of the server on
    which CORBA Naming service is running. */
    // Narrow to root naming context
    NamingContextExt root_context = NamingContextExtHelper.narrow(obj);

}
    catch (org.omg.CORBA.ORBPackage.InvalidName inEx) {
    // Exception handling
}
```

C.2.2.3 Get Reference to EMSSessionFactory

Follow the example in C.2.2.2 Get Reference to the Naming Service, page C-7 to obtain a reference to the naming service.

```
NameComponent name = new NameComponent[6];
name[0] = new NameComponent("TMF_MTNM", "Class");
name[1] = new NameComponent("Cisco Systems", "Vendor");
name[2] = new NameComponent("Cisco Transport Manager", "EMSInstance");
name[3] = new NameComponent(version, "Version"); //where version = "9_1" for CTM 9.2
name[4] = new NameComponent(ctm_sys_id, "EMS"); // ctm_sys_id = "CTM"
name[5] = new NameComponent("SessionFactory", "EmsSessionFactory");
try {
    org.omg.CORBA.Object emsSessionI = root_context.resolve(name);
}
catch (InvalidName inEx) {
    // Exception handling
}
catch (NotFound nfEx) {
    // Exception handling
}
```

C.2.2.4 Implement NmsSession_IOperations

```
import org.tmforum.mtnm.session.*;
import org.tmforum.mtnm.nmsSession.*;

public class SessionImpl implements NmsSession_IOperations {
```

```
    Session_I myAssociatedSession = null;

    public SessionImpl() {
        super();
        // TODO Auto-generated constructor stub
    }

    public void setAssociatedSession(Session_I emsSession) {
         myAssociatedSession = emsSession;
     }
    public Session_I getAssoicatedSession (){ return myAssociatedSession;}

    public void eventLossOccurred(String startTime, String notificationId) {
        // TODO Auto-generated method stub

    }
    public void eventLossCleared(String endTime) {
        // TODO Auto-generated method stub

    }


    public void historyPMDataCompleted(String fileName) {
        // TODO Auto-generated method stub

    }


    public void historyPMDataFailed(String errorReason) {
        // TODO Auto-generated method stub

    }
    public Session_I associatedSession() {
        // TODO Auto-generated method stub
        return null;
    }


    public void ping() {
        // TODO Auto-generated method stub

    }


    public void endSession() {
        // TODO Auto-generated method stub

    }

}
}
```

C.2.2.5 Log In and Retrieve EmsSession

To perform operations from CTM GateWay/CORBA, your client must log in using a username and password created on the CTM client. See B.1 Creating an OSS Client Profile for CTM GateWay/CORBA, page B-1.

```
EmsSession_I m_emsSession = null;
SessionImpl mySessionImpl = new SessionImpl();
try {
    EmsSessionFactory_I ems_ref = EmsSessionFactory_IHelper.narrow(emsSessionI);
```

```
    EmsSession_IHolder emsSessionHldr = new EmsSession_IHolder();
    NmsSession_IPOATie tieobj = new NmsSession_IPOATie(mySessionImpl, poa);
    poa.activate_object(tieobj);
    NmsSession_I nmsSession_ref = tieobj._this();
    if (ems_ref != null) {
        ems_ref.getEmsSession(user, password, nmsSession_ref, emsSessionHldr);
        m_emsSession = emsSessionHldr.value;
    }

    } catch (Exception ex) {
    // System.out.println("Could not narrow");
    ex.printStackTrace();
}
```

C.2.2.6 Retrieve List of Managers

```
managerNames_THolder names = new managerNames_THolder();
m_emsSession.getSupportedManagers(names);
managers = names.value;
for (i = 0; i < managers.length; i++ )
{
    System.out.print("Manager ");
    System.out.print(i);
    System.out.println(" " + managers[i]);
}
```

C.2.2.7 getEMS Operation on EMS Manager

```
EMS_T m_ems;
EMS_THolder m_emsHolder = new EMS_THolder();
try {
    Common_IHolder mgrHolder = new Common_IHolder();
    m_emsSession.getManager("EMS", mgrHolder);
    EMSMgr_I emsMgr = EMSMgr_IHelper.narrow(mgrHolder.value);
    emsMgr.getEMS(m_emsHolder);
}
catch (ProcessingFailureException pfe) {
    System.out.println("Processing Exception" + pfe.getMessage());
    pfe.printStackTrace();
}
m_ems = m_emsHolder.value;
System.out.println("Native EMS Name" + m_ems.nativeEMSName);
```

C.2.2.8 Get Reference to EventChannel

```
EventChannel notifChannel;
EventChannelHolder chanHolder = new EventChannelHolder();
try {
    …
    emsSession.getEventChannel(chanHolder);
}
catch (Exception ex){
    // handle exceptions
}
notifChannel = chanHolder.value;
```

C.2.2.9 Obtain ConsumerAdmin Reference

```
//retrieve default consumer admin
```

```
try {
    ConsumerAdmin cadmin = notifChannel.get_consumeradmin(0);
}
catch (AdminNotFound anfSe) {
    // Exception handling
}
```

C.2.2.10 Obtain ProxyPushSupplier

```
IntHolder id = new IntHolder();
try {
    ProxySupplier baseSupplier =
        cadmin.obtain_notification_push_supplier(
ClientType.STRUCTURED_EVENT, id);
    structuredProxyPushSupplier =
        StructuredProxyPushSupplierHelper.narrow(baseSupplier);
}
catch (AdminLimitExceeded aleEx) {
    // Exception handling
}
```

C.2.2.11 Implement StructuredPushConsumer

```
class StructuredPushConsumerImpl extends _StructuredPushConsumerPOA
{
    StructuredPushConsumerImpl() {
        super();
        System.out.println("StructuredPushConsumerImpl created.");
    }
    public void disconnect_structured_push_consumer() {
        System.out.println("Disconnect structured push consumer.:");
    }
    public void push_structured_event(StructuredEvent notification) {
        System.out.println("Received notification.");
    }
    public void offer_change(EventType[] added,
        EventType[] removed)
        throws InvalidEventType
    {
        System.out.println("Offer changed.");
    }
}
```

C.2.2.12 Connect StructuredPushConsumerImpl

```
try {
    StructuredPushConsumerImpl structProxyPushConsumer = new StructuredPushConsumerImpl();
        StructuredPushConsumerPOATie structuredPushConsumerTieObj = new
        StructuredPushConsumerPOATie (structProxyPushConsumer, poa);
     poa.activate_object(structuredPushConsumerTieObj);
     StructuredPushConsumer pushCon = structuredPushConsumerTieObj._this();
      structuredProxyPushSupplier.connect_structured_push_consumer(pushCon);

    global_orb.run();
}
catch (ServantAlreadyActive sae) {
    // Exception handling
}
catch (WrongPolicy wrongPolicyEx) {
    // Exception handling
}
```

C.2.3 Running the Client

If the initial naming context is not resolved using the Option 2 method shown in C.2.2.2 Get Reference to the Naming Service, page C-7, the following JVM flags must be used for CORBA clients written for Orbix 6.2, Visibroker, or jacORB:

- Orbix 6.2 or later:

  ```
  -ORBInitRef NameService=corbaloc:iiop:1.2@<host name>:<port>/NameServiceGWC
  ```

- Visibroker:

  ```
  -DORBInitRef NameService=corbaloc::<host name>:<port>/NameServiceGWC
  -DORBInitRef NameService=corbaname::<host name>:<port>/NameServiceGWC
  ```

- jacORB:

  ```
  -DORBInitRef.NameService=corbaname::<host name>:<port>/NameServiceGWC
  ```

In addition, if you do not set the system properties org.omg.CORBA.ORBClass and org.omg.CORBA.ORBSingletonClass in your client code before Step 1 in C.2.2.1 Initialize the Client Connection, page C-5, you must set them in your client launch file:

- Orbix 6.2 or later:

  ```
  -Dorg.omg.CORBA.ORBClass=com.iona.corba.art.artimpl.ORBImpl
  -Dorg.omg.CORBA.ORBSingletonClass=com.iona.corba.art.artimpl.ORBSingleton
  ```

- Visibroker:

  ```
  -Dorg.omg.CORBA.ORBClass=com.inprise.vbroker.orb.ORB
  -Dorg.omg.CORBA.ORBSingletonClass=com.inprise.vbroker.orb.ORBSingleton
  ```

- jacORB:

  ```
  -Dorg.omg.CORBA.ORBClass=org.jacorb.orb.ORB
  -Dorg.omg.CORBA.ORBSingletonClass=org.jacorb.orb.ORBSingleton
  ```

For other ORBs, check the related ORB programming guide.

C.2.4 Known Interoperability Issues When Using Orbix

If you use Orbix to set the following property in the jacorb.properties file:

```
Jacorb.compactTypecodes=0
```

Pay attention to the compatibility of the domain you use with Orbix. If you use an insecure domain, set the following property in the jacorb.properties file:

```
Jacorb.security.support_ssl = off
```

APPENDIX D

Equipment List for NEs

This appendix lists the equipment that CTM GateWay/CORBA reports for each NE type. This appendix contains the following sections:

- D.1 Equipment List for MDS 9000, page D-1
- D.2 Equipment List for MGX 8830/8850/8880, page D-2
- D.3 Equipment List for ONS 15216, page D-2
- D.4 Equipment List for ONS 15305, page D-4
- D.5 Equipment List for ONS 15305 CTC, page D-4
- D.6 Equipment List for ONS 15310 CL, page D-5
- D.7 Equipment List for ONS 15310 MA SDH, page D-5
- D.8 Equipment List for ONS 15310 MA SONET, page D-5
- D.9 Equipment List for ONS 15327, page D-6
- D.10 Equipment List for ONS 15454 SDH, page D-6
- D.11 Equipment List for ONS 15454 SONET, page D-9
- D.12 Equipment List for ONS 15454-M2, page D-13
- D.13 Equipment List for ONS 15454-M6, page D-13
- D.14 Equipment List for ONS 15530, page D-13
- D.15 Equipment List for ONS 15540 ESP, page D-26
- D.16 Equipment List for ONS 15540 ESPx, page D-32
- D.17 Equipment List for ONS 15600 SDH, page D-34
- D.18 Equipment List for ONS 15600 SONET, page D-35
- D.19 Equipment List for Unmanaged NEs, page D-36

D.1 Equipment List for MDS 9000

MDS 9000 NEs do not report any equipment.

D.2 Equipment List for MGX 8830/8850/8880

The following table lists the MGX 8830, 8850, and 8880 equipment that CTM R9.2 supports.

Table D-1 CTM-Supported MGX 8830/8850/8880 Equipment

Category	Equipment
Backplane	MGX8880-MG backplane
Multiprotocol Modules	• Multiprotocol service module, 16 T1/E1 ports • Multiprotocol service module, 8 T1/E1 ports • Multiprotocol service module, 3 T3/E3 or one 155 port
PMX modules	• PXM1E, 8 T3/E3 and four 155 broadband ports • PXM1E, 8 T3/E3 broadband ports • PXM1E, four 155 broadband ports • PXM45 Ver C
	Double-height ATM SM, 1 OC-48c/STM-16
PW Supply	Power supply
RPR	RPM-PR
Sensor Cards	• DC level sensor • Fan speed sensor • Temperature sensor • 1.2v sensor • 1.8v sensor • 2.5v sensor • 3.3v sensor • Daughter card temperature sensor 1...8 • Host temperature sensor 1...8
Service Redundancy Modules	Service redundancy module enhanced - B
Voice Switch Cards	• Voice Switch Service Module (VXSM) front card, 4 OC3/STM-1 • VXSM front card, 48 T1/E1 • VXSM front card, 6 T3

D.3 Equipment List for ONS 15216

The equipment is the managed element itself for the following ONS 15216 NEs: ONS 15216 terminal filters, ONS 15216 OADMs, ONS 15216 EDFA, and ONS 15216 OSC.

The following table lists the ONS 15216 equipment that CTM R9.2 supports.

Table D-2 CTM-Supported ONS 15216 Equipment

Card Type	Equipment
DCU	• DCU-100 (100 ps/nm) • DCU-1150 (1150 ps/nm) • DCU-350 (350 ps/nm) • DCU-450 (450 ps/nm) • DCU-550 (550 ps/nm) • DCU-750 (750 ps/nm) • DCU-950 (950 ps/nm) • DCU-DS-L-100 (100 ps/nm) • DCU-DS-L-200 (200 ps/nm) • DCU-DS-L-300 (300 ps/nm) • DCU-E-200 (200 ps/nm, ELEAF type) • DCU-E-350 (350 ps/nm, ELEAF type) • DCU-L-1000 (1000 ps/nm) • DCU-L-1100 (1100 ps/nm) • DCU-L-300 (300 ps/nm) • DCU-L-600 (600 ps/nm) • DCU-L-700 (700 ps/nm) • DCU-L-800 (800 ps/nm)
FLA	FLA-8-*channel-ID*, where *channel-ID* is 36.6, 44.5, 52.5, or 60.6
FLB	FLB-2-*channel-ID*, where *channel-ID* is 31.1, 32.6, 35.0, 36.6, 38.9, 40.5, 42.9, 44.5, 46.9, 48.5, 50.9, 52.5, 54.9, 56.5, 58.9, or 60.6 • CS-2 • CS-3 • CS-4 • CS-MM-Y • CS-SM-Y
SC	SC-4B
VOA	VOA-4

D.4 Equipment List for ONS 15305

The following table lists the ONS 15305 equipment that CTM R9.2 supports.

Table D-3 CTM-Supported ONS 15305 Equipment

Card Type	Equipment
E	• E100-8 • E3T3-6 • E1-8 • E1-21 • E1-63
GE	• GE-1-LC • GE-2-LC
S	• S1.1-8-LC • S4.1-2-LC • S4.1-4-LC • S16.1-1-LC • S1.1-2-LC

D.5 Equipment List for ONS 15305 CTC

The following table lists the ONS 15305 CTC equipment that CTM R9.2 supports.

Table D-4 CTM-Supported ONS 15305 CTC Equipment

Card Type	Equipment
E	• E1-8 • E3T3-6 • E100-8
GE	GigE-WAN-2
S	• S1.1-2-LC • L1.2-2-LC • S1.1-8-LC • S16.1-1-LC • S4.1-2-LC • L16.1-1-LC • S4.1-2-LC

D.6 Equipment List for ONS 15310 CL

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15310 CL user documentation for actual equipment names used by the NE.

Table D-5 CTM-Supported ONS 15310 CL Equipment

Card Type	Equipment
ML	• ML2_L2L3 • ML2_MAPPER
PPM	PPM_1_PORT
XC	CTX

D.7 Equipment List for ONS 15310 MA SDH

The following table lists the ONS 15310 MA SDH equipment that CTM R9.2 supports.

Table D-6 CTM-Supported ONS 15310 MA SDH Equipment

Card Type	Equipment
DS/E	• E1_21_E3_DS3_3 • E1_63_E3_DS3_3
Data Cards	• ML-100T-8 • CE-100T-8 • CE-MR-6
Panel Cards	FILLER

D.8 Equipment List for ONS 15310 MA SONET

The following table lists the ONS 15310 MA SONET equipment that CTM R9.2 supports.

Table D-7 CTM-Supported ONS 15310 MA SONET Equipment

Card Type	Equipment
Controller	CTX-2500
Data Cards	• ML2-100T-8 • CE-100T-8 • CE-MR-6
DS Cards	• DS1-28/DS3-EC1-3 • DS1-84/DS3-EC1-3

Table D-7 CTM-Supported ONS 15310 MA SONET Equipment (continued)

Card Type	Equipment
Fillers	• CTX-FILLER • EXP-FILLER
PPM	PPM_1_PORT

D.9 Equipment List for ONS 15327

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15327 user documentation for actual equipment names used by the NE.

Table D-8 CTM-Supported ONS 15327 Equipment

Card Type	Equipment
ETH	ETH100
G	G1000_2
OC	• OC12 • OC3 • OC48
MIC	XTC
XT	MIC

D.10 Equipment List for ONS 15454 SDH

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15454 SDH user documentation for actual equipment names used by the NE.

Table D-9 CTM-Supported ONS 15454 SDH Equipment

Card Type	Equipment
AD	• AD_1B • AD_1C • AD_2C • AD_4B • AD_4C
ADM	ADM_10G

Table D-9 CTM-Supported ONS 15454 SDH Equipment (continued)

Card Type	Equipment
Aux	• AIC • AICI • AEP • AIE • MMU • PSM • ALM_PWR • CRFT_TMG
CE	• CE-100T-8 • CE-1000-4 • CE-MR-10
Controller	• TCC • TCC2 • TCC2P • TNC • TSC
DS	• DS3I • E1_42 • E3
ETH	• ETH100 • ETH1000
FMEC	• FMEC_SMZ_E1 • FMEC_SMZ_E3 • FMEC_DB • FMEC_SMZ_DS1I
G	• G1000_4
ML	• ML100T • ML1000 • ML-MR-10 • ML100X-8
MR	• MRC-12 • MRC25G_4 • FCMR
MSC	MS-ISC-100T

Table D-9 CTM-Supported ONS 15454 SDH Equipment (continued)

Card Type	Equipment
MXP	• MXP_2.5G_10G • MXP_2.5G_10E • 40G_MXP_C • 10_DME
MUX/DMX	• 40 DMX • 40 MUX • 32DMX • 32DMX_L • 32MUX_O • 32_DMX_O
OPT	• OPT_BST • OPT_PRE • OPT_BST_E • OPT_BST_L • OPT_AML_L • OPT_AMP_17 • OPT_AMP_C • OPT_RAMP_C • OPT_RAMP_CE
OSC	• OSC_CSM • OSCM
SMR	• 40-SMR1-C • 40-SMR2-C
STM	• STM1E_12 • STM1 • STM1_8 • STM4 • STM4_4 • STM16 • STM64 • STM64_XFP
TDCU	• TDCU-C • TDCU-F

Table D-9 CTM-Supported ONS 15454 SDH Equipment (continued)

Card Type	Equipment
TXP	• TXP_MR_10G • TXP_MR_2.5G • TXPP_MR_2.5G • TXP_MR_10E • 2.5G_DM 2.5G_DMP
WSS	• 32WSS • 32WSS_L • 40WSS
WXC	• 40WXC
XC	• XC • XCVT • XC10G • XCVXL_10G • XCVXC • XCVXC25G
XP	• OTU2_XP • 10GE_XP • GE_XP

D.11 Equipment List for ONS 15454 SONET

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15454 SONET user documentation for actual equipment names used by the NE.

Table D-10 CTM-Supported ONS 15454 SONET Equipment

Card Type	Equipment
AD	• AD_1B • AD_1C • AD_2C • AD_4B • AD_4C
ADM	ADM_10G

Table D-10 CTM-Supported ONS 15454 SONET Equipment (continued)

Card Type	Equipment
Aux	• AIC • AICI • AEP • AIE • MMU • PSM • ALM_PWR • CRFT_TMG
CE	• CE-100T-8 • CE-1000-4 • CE-MR-10
Controller	• TCC • TCC2 • TCC2P • TNC • TSC
DS	• DS1 • DS1N • DS1-63 • DS3 • DS3N • DS3E • DS3NE • DS3XM • DS3XM_12 • DS3I • DS3IN • DS3_EC1_48 • DS1_E1_56 • EC1
ETH	• ETH100 • ETH1000

Table D-10 CTM-Supported ONS 15454 SONET Equipment (continued)

Card Type	Equipment
FMEC	• FMEC_SMZ_E1 • FMEC_SMZ_E3 • FMEC_DB • FMEC_SMZ_DS1I
G	G1000_4
ML	• ML100T • ML1000 • ML-MR-10 • ML100X-8
MR	• MRC-12 • MRC25G_4 • FCMR
MSC	MS-ISC-100T
MXP	• MXP_2.5G_10G • MXP_2.5G_10E • 40G_MXP_C • 10_DME
MUX/DMX	• 40 DMX • 40 MUX • 32DMX • 32DMX_L • 32MUX_O • 32_DMX_O
OC	• OC3 • OC3_8 • OC12 • OC12_4 • OC48 • OC192 • OC192_XFP

Table D-10 CTM-Supported ONS 15454 SONET Equipment (continued)

Card Type	Equipment
OPT	• OPT_BST • OPT_PRE • OPT_BST_E • OPT_BST_L • OPT_AML_L • OPT_AMP_17 • OPT_AMP_C • OPT_RAMP_C • OPT_RAMP_CE
OSC	• OSC_CSM • OSCM
SMR	• 40-SMR1-C • 40-SMR2-C
TDCU	• TDCU-C • TDCU-F
TXP	• TXP_MR_10G • TXP_MR_2.5G • TXPP_MR_2.5G • TXP_MR_10E • 2.5G_DM 2.5G_DMP
WSS	• 32WSS • 32WSS_L • 40WSS
WXC	40WXC
XC	• XC • XCVT • XC10G • XCVXL_10G • XCVXC • XCVXC25G
XP	• OTU2_XP • 10GE_XP • GE_XP

D.12 Equipment List for ONS 15454-M2

The ONS 15454-M2 supports all equipment that is supported by the ONS 15454 MSTP system.

Note ONS 15454-M2 does not support OSCM, AICs, EOL units, or ancillary units.

D.13 Equipment List for ONS 15454-M6

The ONS 15454-M6 supports all equipment that is supported by the ONS 15454 MSTP system.

Note ONS 15454-M6 does not support OSCM, AIC, EOL units, or ancillary units.

D.14 Equipment List for ONS 15530

The equipment name is reported by the actual managed element. The following table lists the ONS 15530 equipment that CTM R9.2 supports.

Table D-11 CTM-Supported ONS 15530 Equipment

Equipment Name	Equipment Contained In	Description
Chassis		
cevChassisONS15530NEBS	0	Cisco ONS 15530 chassis, NEBS-compliant
cevChassisONS15530ETSI	0	Cisco ONS 15530 chassis, ETSI-compliant
Container Entities		
cevContainerONS15530CardSlot	cevChassisONS15530	Cisco ONS 15530 processor, transponder, or any line card module slot
cevContainerONS15530FanTraySlot	cevChassisONS15530	Cisco ONS 15530 fan tray slot
cevContainerONS15530PSSlot	cevChassisONS15530	Cisco ONS 15530 power supply slot
cevContainerONS15530SubCardSlot	cevONS15530LCMBOsc, …	Cisco ONS 15530 subslot for OSC module
cevContainerONS15530PortSlot	cevONS15530LCEscon10, …	Cisco ONS 15530 slot for transceiver
Power Supply		
cevPowerSupply15530AC	cevContainerONS15530PSSlot	Cisco ONS 15530 120-240 VAC power supply
cevPowerSupply15530DC	cevContainerONS15530PSSlot	Cisco ONS 15530 -48 VDC power supply
cevFanONS15530FanTray6	cevContainerONS15530FanTraySlot	Cisco ONS 15530 fan tray module with six fans
Processor Switch Card		
cevCpuONS15530SW	cevContainerONS15530CardSlot	Cisco ONS 15530 CPU and switch board
OSC Motherboard		

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530LCMBOsc	cevContainerONS15530CardSlot	ONS 15530 optical supervisor channel line card motherboard
OSC Subcard		
cevONS15530Osc	cevContainerONS15530SubCardSlot	ONS 15530 optical supervisor channel module
Mux/Demux Subcards Without OSC		
cevONS15530MDX4ChA	cevContainerONS15530CardSlot	ONS 15530 4-channel band A mux/demux module, without OSC
cevONS15530MDX4ChB	cevContainerONS15530CardSlot	ONS 15530 4-channel band B mux/demux module, without OSC
cevONS15530MDX4ChC	cevContainerONS15530CardSlot	ONS 15530 4-channel band C mux/demux module, without OSC
cevONS15530MDX4ChD	cevContainerONS15530CardSlot	ONS 15530 4-channel band D mux/demux module, without OSC
cevONS15530MDX4ChE	cevContainerONS15530CardSlot	ONS 15530 4-channel band E mux/demux module, without OSC
cevONS15530MDX4ChF	cevContainerONS15530CardSlot	ONS 15530 4-channel band F mux/demux module, without OSC
cevONS15530MDX4ChG	cevContainerONS15530CardSlot	ONS 15530 4-channel band G mux/demux module, without OSC
cevONS15530MDX4ChH	cevContainerONS15530CardSlot	ONS 15530 4-channel band H mux/demux module, without OSC
Mux/Demux Subcards with OSC		
cevONS15530MDX4ChAOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band A mux/demux module, with OSC
cevONS15530MDX4ChBOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band B mux/demux module, with OSC
cevONS15530MDX4ChCOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band C mux/demux module, with OSC
cevONS15530MDX4ChDOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band D mux/demux module, with OSC
cevONS15530MDX4ChEOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band E mux/demux module, with OSC
cevONS15530MDX4ChFOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band F mux/demux module, with OSC
cevONS15530MDX4ChGOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band G mux/demux module, with OSC
cevONS15530MDX4ChHOsc	cevContainerONS15530CardSlot	ONS 15530 4-channel band H mux/demux module, with OSC
LBR Transponder with Splitter		
cevONS15530Tsp1Ch0102MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 1 or 2, 1310-nm MM, SC connector, with splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530Tsp1Ch0304MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 3 or 4, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch0506MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 5 or 6, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch0708MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 7 or 8, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch0910MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 9 or 10, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch1112MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 11 or 12, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch1314MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 13 or 14, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch1516MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 15 or 16, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch1718MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 17 or 18, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch1920MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 19 or 20, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch2122MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 21 or 22, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch2324MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 23 or 24, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch2526MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 25 or 26, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch2728MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 27 or 28, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch2930MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 29 or 30, 1310-nm MM, SC connector, with splitter
cevONS15530Tsp1Ch3132MmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 31 or 32, 1310-nm MM, SC connector, with splitter
Transponder with Splitter		

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530Tsp1Ch0102SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 1 or 2, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch0304SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 3 or 4, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch0506SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 5 or 6, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch0708SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 7 or 8, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch0910SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 9 or 10, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch1112SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 11 or 12, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch1314SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 13 or 14, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch1516SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 15 or 16, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch1718SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 17 or 18, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch1920SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 19 or 20, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch2122SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 21 or 22, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch2324SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 23 or 24, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch2526SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 25 or 26, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch2728SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 27 or 28, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch2930SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 29 or 30, 1310-nm SM, SC connector, with splitter
cevONS15530Tsp1Ch3132SmScSp	cevContainerONS15530CardSlot	Transponder type I line card, channel 31 or 32, 1310-nm SM, SC connector, with splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530Tsp1Ch0102MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 1 or 2, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch0304MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 3 or 4, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch0506MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 5 or 6, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch0708MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 7 or 8, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch0910MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 9 or 10, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch1112MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 11 or 12, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch1314MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 13 or 14, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch1516MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 15 or 16, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch1718MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 17 or 18, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch1920MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 19 or 20, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch2122MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 21 or 22, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch2324MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 23 or 24, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch2526MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 25 or 26, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch2728MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 27 or 28, 1310-nm MM, SC connector, without splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530Tsp1Ch2930MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 29 or 30, 1310-nm MM, SC connector, without splitter
cevONS15530Tsp1Ch3132MmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 31 or 32, 1310-nm MM, SC connector, without splitter
Transponder Without Splitter		
cevONS15530Tsp1Ch0102SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 1 or 2, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch0304SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 3 or 4, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch0506SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 5 or 6, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch0708SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 7 or 8, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch0910SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 9 or 10, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch1112SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 11 or 12, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch1314SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 13 or 14, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch1516SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 15 or 16, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch1718SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 17 or 18, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch1920SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 19 or 20, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch2122SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 21 or 22, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch2324SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 23 or 24, 1310-nm SM, SC connector, without splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15530Tsp1Ch2526SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 25 or 26, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch2728SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 27 or 28, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch2930SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 29 or 30, 1310-nm SM, SC connector, without splitter
cevONS15530Tsp1Ch3132SmSc	cevContainerONS15530CardSlot	Transponder type I line card, channel 31 or 32, 1310-nm SM, SC connector, without splitter
ESCON Line Card (10 Port)		
cevONS15530LCEscon10	cevContainerONS15530CardSlot	ONS 15530 10-port ESCON multiplexing line card
ESCON SFP		
cevONS15500XvrEscon1310MmMTRJ (ignore this vendor type. 15530 uses cevONS15500XvrEscon1310MTRJ)	cevContainerONS15530PortSlot	ONS 15500 pluggable transceiver ESCON, 1310 nm MM, MTRJ
10 G Uplink Card		
cevONS15530LC10GESmSc	cevContainerONS15530CardSlot	ONS 15530 10-Gbps uplink, 1310 nm, with SC
10 G Trunk, ITU Module, with Splitter		
cevONS1553010GCh01Sp	cevContainerONS15530CardSlot	ONS 15530 channel 01, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh02Sp	cevContainerONS15530CardSlot	ONS 15530 channel 02, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh03Sp	cevContainerONS15530CardSlot	ONS 15530 channel 03, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh04Sp	cevContainerONS15530CardSlot	ONS 15530 channel 04, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh05Sp	cevContainerONS15530CardSlot	ONS 15530 channel 05, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh06Sp	cevContainerONS15530CardSlot	ONS 15530 channel 06, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh07Sp	cevContainerONS15530CardSlot	ONS 15530 channel 07, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh08Sp	cevContainerONS15530CardSlot	ONS 15530 channel 08, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh09Sp	cevContainerONS15530CardSlot	ONS 15530 channel 09, 10-Gbps ITU line card MU, with splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS1553010GCh10Sp	cevContainerONS15530CardSlot	ONS 15530 channel 10, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh11Sp	cevContainerONS15530CardSlot	ONS 15530 channel 11, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh12Sp	cevContainerONS15530CardSlot	ONS 15530 channel 12, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh13Sp	cevContainerONS15530CardSlot	ONS 15530 channel 13, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh14Sp	cevContainerONS15530CardSlot	ONS 15530 channel 14, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh15Sp	cevContainerONS15530CardSlot	ONS 15530 channel 15, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh16Sp	cevContainerONS15530CardSlot	ONS 15530 channel 16, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh17Sp	cevContainerONS15530CardSlot	ONS 15530 channel 17, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh18Sp	cevContainerONS15530CardSlot	ONS 15530 channel 18, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh19Sp	cevContainerONS15530CardSlot	ONS 15530 channel 19, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh10Sp	cevContainerONS15530CardSlot	ONS 15530 channel 20, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh21Sp	cevContainerONS15530CardSlot	ONS 15530 channel 21, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh22Sp	cevContainerONS15530CardSlot	ONS 15530 channel 22, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh23Sp	cevContainerONS15530CardSlot	ONS 15530 channel 23, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh24Sp	cevContainerONS15530CardSlot	ONS 15530 channel 24, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh25Sp	cevContainerONS15530CardSlot	ONS 15530 channel 25, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh26Sp	cevContainerONS15530CardSlot	ONS 15530 channel 26, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh27Sp	cevContainerONS15530CardSlot	ONS 15530 channel 27, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh28Sp	cevContainerONS15530CardSlot	ONS 15530 channel 28, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh29Sp	cevContainerONS15530CardSlot	ONS 15530 channel 29, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh30Sp	cevContainerONS15530CardSlot	ONS 15530 channel 30, 10-Gbps ITU line card MU, with splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS1553010GCh31Sp	cevContainerONS15530CardSlot	ONS 15530 channel 31, 10-Gbps ITU line card MU, with splitter
cevONS1553010GCh32Sp	cevContainerONS15530CardSlot	ONS 15530 channel 32, 10-Gbps ITU line card MU, with splitter
10 G Trunk, ITU Module, Without Splitter		
cevONS1553010GCh01	cevContainerONS15530CardSlot	ONS 15530 channel 01, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh02	cevContainerONS15530CardSlot	ONS 15530 channel 02, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh03	cevContainerONS15530CardSlot	ONS 15530 channel 03, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh04	cevContainerONS15530CardSlot	ONS 15530 channel 04, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh05	cevContainerONS15530CardSlot	ONS 15530 channel 05, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh06	cevContainerONS15530CardSlot	ONS 15530 channel 06, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh07	cevContainerONS15530CardSlot	ONS 15530 channel 07, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh08	cevContainerONS15530CardSlot	ONS 15530 channel 08, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh09	cevContainerONS15530CardSlot	ONS 15530 channel 09, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh10	cevContainerONS15530CardSlot	ONS 15530 channel 10, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh11	cevContainerONS15530CardSlot	ONS 15530 channel 11, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh12	cevContainerONS15530CardSlot	ONS 15530 channel 12, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh13	cevContainerONS15530CardSlot	ONS 15530 channel 13, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh14	cevContainerONS15530CardSlot	ONS 15530 channel 14, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh15	cevContainerONS15530CardSlot	ONS 15530 channel 15, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh16	cevContainerONS15530CardSlot	ONS 15530 channel 16, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh17	cevContainerONS15530CardSlot	ONS 15530 channel 17, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh18	cevContainerONS15530CardSlot	ONS 15530 channel 18, 10-Gbps ITU line card MU, without splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS1553010GCh19	cevContainerONS15530CardSlot	ONS 15530 channel 19, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh10	cevContainerONS15530CardSlot	ONS 15530 channel 20, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh21	cevContainerONS15530CardSlot	ONS 15530 channel 21, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh22	cevContainerONS15530CardSlot	ONS 15530 channel 22, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh23	cevContainerONS15530CardSlot	ONS 15530 channel 23, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh24	cevContainerONS15530CardSlot	ONS 15530 channel 24, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh25	cevContainerONS15530CardSlot	ONS 15530 channel 25, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh26	cevContainerONS15530CardSlot	ONS 15530 channel 26, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh27	cevContainerONS15530CardSlot	ONS 15530 channel 27, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh28	cevContainerONS15530CardSlot	ONS 15530 channel 28, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh29	cevContainerONS15530CardSlot	ONS 15530 channel 29, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh30	cevContainerONS15530CardSlot	ONS 15530 channel 30, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh31	cevContainerONS15530CardSlot	ONS 15530 channel 31, 10-Gbps ITU line card MU, without splitter
cevONS1553010GCh32	cevContainerONS15530CardSlot	ONS 15530 channel 32, 10-Gbps ITU line card MU, without splitter
8-Port Fiber Channel/Gigabit Ethernet Line Card		
cevONS15530FCGE8P	cevContainerONS15530CardSlot	ONS 15530 8-port fiber channel/Gigabit Ethernet line card
2.5 G, ITU Module, with Splitter		
cevONS155302p5GCh0102Sp	cevContainerONS15530CardSlot	ONS 15530 channel 01 or 02, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh0304Sp	cevContainerONS15530CardSlot	ONS 15530 channel 03 or 04, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh0506Sp	cevContainerONS15530CardSlot	ONS 15530 channel 05 or 06, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh0708Sp	cevContainerONS15530CardSlot	ONS 15530 channel 07 or 08, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh0910Sp	cevContainerONS15530CardSlot	ONS 15530 channel 09 or 10, 2.5-Gbps ITU line card MU, with splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS155302p5GCh1112Sp	cevContainerONS15530CardSlot	ONS 15530 channel 11 or 12, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh1314Sp	cevContainerONS15530CardSlot	ONS 15530 channel 13 or 14, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh1516Sp	cevContainerONS15530CardSlot	ONS 15530 channel 15 or 16, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh1718Sp	cevContainerONS15530CardSlot	ONS 15530 channel 17 or 18, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh1920Sp	cevContainerONS15530CardSlot	ONS 15530 channel 19 or 20, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh2122Sp	cevContainerONS15530CardSlot	ONS 15530 channel 21 or 22, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh2324Sp	cevContainerONS15530CardSlot	ONS 15530 channel 23 or 24, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh2526Sp	cevContainerONS15530CardSlot	ONS 15530 channel 25 or 26, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh2728Sp	cevContainerONS15530CardSlot	ONS 15530 channel 27 or 28, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh2930Sp	cevContainerONS15530CardSlot	ONS 15530 channel 29 or 30, 2.5-Gbps ITU line card MU, with splitter
cevONS155302p5GCh3132Sp	cevContainerONS15530CardSlot	ONS 15530 channel 31 or 32, 2.5-Gbps ITU line card MU, with splitter
2.5 G, ITU Module, Without Splitter		
cevONS155302p5GCh0102	cevContainerONS15530CardSlot	ONS 15530 channel 01 or 02, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh0304	cevContainerONS15530CardSlot	ONS 15530 channel 03 or 04, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh0506	cevContainerONS15530CardSlot	ONS 15530 channel 05 or 06, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh0708	cevContainerONS15530CardSlot	ONS 15530 channel 07 or 08, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh0910	cevContainerONS15530CardSlot	ONS 15530 channel 09 or 10, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh1112	cevContainerONS15530CardSlot	ONS 15530 channel 11 or 12, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh1314	cevContainerONS15530CardSlot	ONS 15530 channel 13 or 14, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh1516	cevContainerONS15530CardSlot	ONS 15530 channel 15 or 16, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh1718	cevContainerONS15530CardSlot	ONS 15530 channel 17 or 18, 2.5-Gbps ITU line card MU, without splitter

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS155302p5GCh1920	cevContainerONS15530CardSlot	ONS 15530 channel 19 or 20, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh2122	cevContainerONS15530CardSlot	ONS 15530 channel 21 or 22, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh2324	cevContainerONS15530CardSlot	ONS 15530 channel 23 or 24, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh2526	cevContainerONS15530CardSlot	ONS 15530 channel 25 or 26, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh2728	cevContainerONS15530CardSlot	ONS 15530 channel 27 or 28, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh2930	cevContainerONS15530CardSlot	ONS 15530 channel 29 or 30, 2.5-Gbps ITU line card MU, without splitter
cevONS155302p5GCh3132	cevContainerONS15530CardSlot	ONS 15530 channel 31 or 32, 2.5-Gbps ITU line card MU, without splitter
Ports		
cevPortWaveOsc	cevONS15530Osc	Optical port that transmits and receives one ITU DWDM grid wavelength, used as the optical supervisory channel
cevPortWaveEthPhy	—	Optical port that transports Ethernet packets in the ITU DWDM grid wavelength
cevPort10GigEthPhy	—	Optical port that transports 10-Gb Ethernet packets
cevPortEsconPhy	—	Optical port that transports ESCON traffic
cevPortGEFCPhy	—	Optical port that transports Gigabit Ethernet and fibre channel traffic
cevPortVOAIn	cevONS15500VOA*	Input port on a variable optical attenuator module
cevPortVOAOut	cevONS15500VOA*	Output port on a variable optical attenuator module
cevPortVOAFilterIn	cevONS15500PEQ*	Input port on an optical power equalizer module that filters and attenuates a selective band of wavelengths
cevPortVOAFilterOut	cevONS15500PEQ*	Output port on an optical power equalizer module that filters and attenuates a selective band of wavelengths
cevPortVOAByPassIn	cevONS15500PEQ*	Port on an optical power equalizer module that accepts DWDM wavelengths that are not attenuated but must be combined with the output signal
cevPortVOAByPassOut	cevONS15500PEQ*	Port on an optical power equalizer module that outputs those DWDM wavelengths in the input signal that are not attenuated

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
Single-Band Power Equalizer Module		
cevONS15500PEQ01A	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band A
cevONS15500PEQ01B	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band B
cevONS15500PEQ01C	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band C
cevONS15500PEQ01D	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band D
cevONS15500PEQ01E	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band E
cevONS15500PEQ01F	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band F
cevONS15500PEQ01G	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band G
cevONS15500PEQ01H	cevContainerONS15530SubCardSlot	ONS 15500 single-band power equalizer, band H
Dual-Band Power Equalizer Module		
cevONS15500PEQ02AB	cevContainerONS15530SubCardSlot	ONS 15500 dual-band power equalizer, band AB
cevONS15500PEQ02CD	cevContainerONS15530SubCardSlot	ONS 15500 dual-band power equalizer, band CD
cevONS15500PEQ02EF	cevContainerONS15530SubCardSlot	ONS 15500 dual-band power equalizer, band EF
cevONS15500PEQ02GH	cevContainerONS15530SubCardSlot	ONS 15500 dual-band power equalizer, band GH
Variable Optical Attenuator (VOA)		
cevONS15500VOA01	cevContainerONS15530SubCardSlot	ONS 15500, one variable optical attenuator
cevONS15500VOA02	cevContainerONS15530SubCardSlot	ONS 15500, two variable optical attenuators
Pluggable Transceiver Modules for Type 2 Transponder		
cevONS15500XvrEscon1310MTRJ	cevContainerONS155xxPortSlot	ESCON, 1310 nm MM, MTRJ
cevONS15500XvrEscon1310LC	cevContainerONS155xxPortSlot	ESCON, 1310 nm MM, LC
cevONS15500XvrFC850LC	cevContainerONS155xxPortSlot	Fibre channel, 850 nm MM, LC
cevONS15500XvrFC850MTRJ	cevContainerONS155xxPortSlot	Fibre channel, 850 nm MM, MTRJ
cevONS15500XvrFC1310LC	cevContainerONS155xxPortSlot	Fibre channel, 1310 nm SM, LC
cevONS15500XvrFC1310MTRJ	cevContainerONS155xxPortSlot	Fibre channel, 1310 nm SM, MTRJ
cevONS15500XvrGE850LC	cevContainerONS155xxPortSlot	Gigabit Ethernet, 850 nm MM, LC
cevONS15500XvrGE850MTRJ	cevContainerONS155xxPortSlot	Gigabit Ethernet, 850 nm MM, MTRJ
cevONS15500XvrGE1310LC	cevContainerONS155xxPortSlot	Gigabit Ethernet, 1310 nm SM, LC

Table D-11 CTM-Supported ONS 15530 Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15500XvrGE1310MTRJ	cevContainerONS155xxPortSlot	Gigabit Ethernet, 1310 nm SM, MTRJ
cevONS15500XvrGEFC850LC	cevContainerONS155xxPortSlot	Gigabit Ethernet, fibre channel, 850 nm MM, LC
cevONS15500XvrGEFC850MTRJ	cevContainerONS155xxPortSlot	Gigabit Ethernet, fibre channel, 850 nm MM, MTRJ
cevONS15500XvrGEFC1310LC	cevContainerONS155xxPortSlot	Gigabit Ethernet, fibre channel, 1310 nm SM, LC
cevONS15500XvrGEFC1310MTRJ	cevContainerONS155xxPortSlot	Gigabit Ethernet, fibre channel, 1310 nm SM, MTRJ
cevONS15500XvrOC3Son1310MTRJ	cevContainerONS155xxPortSlot	SONET OC3, 1310 nm SM, MTRJ
cevONS15500XvrOC3Son1310LC	cevContainerONS155xxPortSlot	SONET OC3, 1310 nm SM, LC
cevONS15500XvrOC12Son1310MTRJ	cevContainerONS155xxPortSlot	SONET OC12, 1310 nm SM, MTRJ
cevONS15500XvrOC12Son1310LC	cevContainerONS155xxPortSlot	SONET OC12, 1310 nm SM, LC
cevONS15500XvrOC48Son1310MTRJ	cevContainerONS155xxPortSlot	SONET OC48, 1310 nm SM, MTRJ
cevONS15500XvrOC48Son1310LC	cevContainerONS155xxPortSlot	SONET OC48, 1310 nm SM, LC

D.15 Equipment List for ONS 15540 ESP

The equipment name is reported by the actual managed element. The following table lists the ONS 15540 Extended Services Platform (ESP) equipment that CTM R9.2 supports.

Table D-12 CTM-Supported ONS 15540 ESP Equipment

Equipment Name	Equipment Contained In	Description
cevChassisONS15540CHSA	—	Chassis
cevFanONS15540FanTray8	cevContainerONS15540FanTraySlot	ONS15540 fan tray module populated with eight fans
Modules		
cevModule15540LCMBTspWithProtect	CevContainerONS15540CardSlot	Line card motherboard for four type 1 and 2 transponders, with protection
cevModule15540LCMBTspEast	CevContainerONS15540CardSlot	Line card motherboard for four type 1 and 2 transponders, without protection, east
cevModule15540LCMBTspWest	cevContainerONS15540CardSlot	Line card motherboard for four type 1 and 2 transponders, without protection, west
cevONS15540LCMB10GOC192c	cevContainerONS15540CardSlot	ONS 15540 ESP, supports two 10GE/OC192c modules, without splitter
cevModule15540MuxDmxMBWithOSC	cevContainerONS15540CardSlot	Mux/demux motherboard, with OSC

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevModule15540MuxDmxMBWoOSC	cevContainerONS15540CardSlot	Mux/demux motherboard, without OSC
cevCpuONS15540	cevContainerONS15540CardSlot	ONS15540 CPU board, without switch fabric
cevONS15540LCDC10G	cevContainerONS15540SubCardSlot	ONS 15540 ESP, 10GE/OC192 adaptor daughter card
cevONS15540Tsp1Ch0102MMSC	cevModule15540LCMBx	Transponder type I, channel 1/2, 1310 nm MM, SC
cevONS15540Tsp1Ch0304MMSC	cevModule15540LCMBx	Transponder type I, channel 3/4, 1310 nm MM, SC
cevONS15540Tsp1Ch0506MMSC	cevModule15540LCMBx	Transponder type I, channel 5/6, 1310 nm MM, SC
cevONS15540Tsp1Ch0708MMSC	cevModule15540LCMBx	Transponder type I, channel 7/8, 1310 nm MM, SC
cevONS15540Tsp1Ch0910MMSC	cevModule15540LCMBx	Transponder type I, channel 9/10, 1310 nm MM, SC
cevONS15540Tsp1Ch1112MMSC	cevModule15540LCMBx	Transponder type I, channel 11/12, 1310 nm MM, SC
cevONS15540Tsp1Ch1314MMSC	cevModule15540LCMBx	Transponder type I, channel 13/14, 1310 nm MM, SC
cevONS15540Tsp1Ch1516MMSC	cevModule15540LCMBx	Transponder type I, channel 15/16, 1310 nm MM, SC
cevONS15540Tsp1Ch1718MMSC	cevModule15540LCMBx	Transponder type I, channel 17/18, 1310 nm MM, SC
cevONS15540Tsp1Ch1920MMSC	cevModule15540LCMBx	Transponder type I, channel 19/20, 1310 nm MM, SC
cevONS15540Tsp1Ch2122MMSC	cevModule15540LCMBx	Transponder type I, channel 21/22, 1310 nm MM, SC
cevONS15540Tsp1Ch2324MMSC	cevModule15540LCMBx	Transponder type I, channel 23/24, 1310 nm MM, SC
cevONS15540Tsp1Ch2526MMSC	cevModule15540LCMBx	Transponder type I, channel 25/26, 1310 nm MM, SC
cevONS15540Tsp1Ch2728MMSC	cevModule15540LCMBx	Transponder type I, channel 27/28, 1310 nm MM, SC
cevONS15540Tsp1Ch2930MMSC	cevModule15540LCMBx	Transponder type I, channel 29/30, 1310 nm MM, SC
cevONS15540Tsp1Ch3132MMSC	cevModule15540LCMBx	Transponder type I, channel 31/32, 1310 nm MM, SC
cevONS15540Tsp1Ch0102SMSC	cevModule15540LCMBx	Transponder type I, channel 1/2, 1310 nm SM, SC
cevONS15540Tsp1Ch0304SMSC	cevModule15540LCMBx	Transponder type I, channel 3/4, 1310 nm SM, SC

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15540Tsp1Ch0506SMSC	cevModule15540LCMBx	Transponder type I, channel 5/6, 1310 nm SM, SC
cevONS15540Tsp1Ch0708SMSC	cevModule15540LCMBx	Transponder type I, channel 7/8, 1310 nm SM, SC
cevONS15540Tsp1Ch0910SMSC	cevModule15540LCMBx	Transponder type I, channel 9/10, 1310 nm SM, SC
cevONS15540Tsp1Ch1112SMSC	cevModule15540LCMBx	Transponder type I, channel 11/12, 1310 nm SM, SC
cevONS15540Tsp1Ch1314SMSC	cevModule15540LCMBx	Transponder type I, channel 13/14, 1310 nm SM, SC
cevONS15540Tsp1Ch1516SMSC	cevModule15540LCMBx	Transponder type I, channel 15/16, 1310 nm SM, SC
cevONS15540Tsp1Ch1718SMSC	cevModule15540LCMBx	Transponder type I, channel 17/18, 1310 nm SM, SC
cevONS15540Tsp1Ch1920SMSC	cevModule15540LCMBx	Transponder type I, channel 19/20, 1310 nm SM, SC
cevONS15540Tsp1Ch2122SMSC	cevModule15540LCMBx	Transponder type I, channel 21/22, 1310 nm SM, SC
cevONS15540Tsp1Ch2324SMSC	cevModule15540LCMBx	Transponder type I, channel 23/24, 1310 nm SM, SC
cevONS15540Tsp1Ch2526SMSC	cevModule15540LCMBx	Transponder type I, channel 25/26, 1310 nm SM, SC
cevONS15540Tsp1Ch2728SMSC	cevModule15540LCMBx	Transponder type I, channel 27/28, 1310 nm SM, SC
cevONS15540Tsp1Ch2930SMSC	cevModule15540LCMBx	Transponder type I, channel 29/30, 1310 nm SM, SC
cevONS15540Tsp1Ch3132SMSC	cevModule15540LCMBx	Transponder type I, channel 31/32, 1310 nm SM, SC
cevONS15540MDX4ChA	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band A, without OSC
cevONS15540MDX4ChB	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band B, without OSC
cevONS15540MDX4ChC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band C, without OSC
cevONS15540MDX4ChD	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band D, without OSC
cevONS15540MDX4ChE	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band E, without OSC
cevONS15540MDX4ChF	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band F, without OSC
cevONS15540MDX4ChG	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band G, without OSC

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15540MDX4ChH	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band H, without OSC
cevONS15540MDX8ChAB	cevONS15540MDXMBx	8-channel optical add/drop mux/demux, band AB, without OSC
cevONS15540MDX8ChCD	cevONS15540MDXMBx	8-channel optical add/drop mux/demux, band CD, without OSC
cevONS15540MDX8ChEF	cevONS15540MDXMBx	8-channel optical add/drop mux/demux, band EF, without OSC
cevONS15540MDX8ChGH	cevONS15540MDXMBx	8-channel optical add/drop mux/demux, band GH, without OSC
cevONS15540MDX16ChEH	cevONS15540MDXMBx	16-channel optical terminal mux/demux, band EH, without OSC
cevONS15540MDX4ChAOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band A, with OSC
cevONS15540MDX4ChBOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band B, with OSC
cevONS15540MDX4ChCOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band C, with OSC
cevONS15540MDX4ChDOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band D, with OSC
cevONS15540MDX4ChEOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band E, with OSC
cevONS15540MDX4ChFOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band F, with OSC
cevONS15540MDX4ChGOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band G, with OSC
cevONS15540MDX4ChHOSC	cevONS15540MDXMBx	4-channel optical add/drop mux/demux, band H, with OSC
cevONS15540MDX8ChABOSC	cevONS15540MDXMBOSC	8-channel optical add/drop mux/demux, band AB, with OSC
cevONS15540MDX8ChCDOSC	cevONS15540MDXMBOSC	8-channel optical add/drop mux/demux, band CD, with OSC
cevONS15540MDX8ChEFOSC	cevONS15540MDXMBOSC	8-channel optical add/drop mux/demux, band EF, with OSC
cevONS15540MDX8ChGHOSC	cevONS15540MDXMBOSC	8-channel optical add/drop mux/demux, band GH, with OSC
cevONS15540MDX16ChADOSC	cevONS15540MDXMBOSC	16-channel optical terminal mux/demux, band AD, with OSC
Type II Transponder Module with 2-Channel ITU Tunable Lasers		
cevONS15540Tsp2Ch0102	cevContainerONS15540SubCardSlot	Transponder type II, channel 1/2
cevONS15540Tsp2Ch0304	cevContainerONS15540SubCardSlot	Transponder type II, channel 3/4

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS15540Tsp2Ch0506	cevContainerONS15540SubCardSlot	Transponder type II, channel 5/6
cevONS15540Tsp2Ch0708	cevContainerONS15540SubCardSlot	Transponder type II, channel 7/8
cevONS15540Tsp2Ch0910	cevContainerONS15540SubCardSlot	Transponder type II, channel 9/10
cevONS15540Tsp2Ch1112	cevContainerONS15540SubCardSlot	Transponder type II, channel 11/12
cevONS15540Tsp2Ch1314	cevContainerONS15540SubCardSlot	Transponder type II, channel 13/14
cevONS15540Tsp2Ch1516	cevContainerONS15540SubCardSlot	Transponder type II, channel 15/16
cevONS15540Tsp2Ch1718	cevContainerONS15540SubCardSlot	Transponder type II, channel 17/18
cevONS15540Tsp2Ch1920	cevContainerONS15540SubCardSlot	Transponder type II, channel 19/20
cevONS15540Tsp2Ch2122	cevContainerONS15540SubCardSlot	Transponder type II, channel 21/22
cevONS15540Tsp2Ch2324	cevContainerONS15540SubCardSlot	Transponder type II, channel 23/24
cevONS15540Tsp2Ch2425	cevContainerONS15540SubCardSlot	Transponder type II, channel 25/26
cevONS15540Tsp2Ch2628	cevContainerONS15540SubCardSlot	Transponder type II, channel 27/28
cevONS15540Tsp2Ch2930	cevContainerONS15540SubCardSlot	Transponder type II, channel 29/30
cevONS15540Tsp2Ch3132	cevContainerONS15540SubCardSlot	Transponder type II, channel 31/32
Pluggable Transceiver Modules for Type 2 Transponder		
cevONS15540XvrEscon1310MTRJ	cevContainerONS15540PortSlot	ESCON, 1310 nm MM, MTRJ
cevONS15540XvrEscon1310MTLC	cevContainerONS15540PortSlot	ESCON, 1310 nm MM, MTLC
cevONS15540XvrFC850MTLC	cevContainerONS15540PortSlot	Fibre channel, 850 nm MM, MTLC
cevONS15540XvrFC850MTRJ	cevContainerONS15540PortSlot	Fibre channel, 850 nm MM, MTRJ
cevONS15540XvrFC1310MTLC	cevContainerONS15540PortSlot	Fibre channel, 1310 nm SM, MTLC
cevONS15540XvrFC1310MTRJ	cevContainerONS15540PortSlot	Fibre channel, 1310 nm SM, MTRJ
cevONS15540XvrGE850MTLC	cevContainerONS15540PortSlot	Gigabit Ethernet, 850 nm MM, MTLC
cevONS15540XvrGE850MTRJ	cevContainerONS15540PortSlot	Gigabit Ethernet, 850 nm MM, MTRJ
cevONS15540XvrGE1310MTLC	cevContainerONS15540PortSlot	Gigabit Ethernet, 1310 nm SM, MTLC
cevONS15540XvrGE1310MTRJ	cevContainerONS15540PortSlot	Gigabit Ethernet, 1310 nm SM, MTRJ
cevONS15540XvrGEFC850MTLC	cevContainerONS15540PortSlot	Gigabit Ethernet, fibre channel, 850 nm MM, MTLC
cevONS15540XvrGEFC850MTRJ	cevContainerONS15540PortSlot	Gigabit Ethernet, fibre channel, 850 nm MM, MTRJ
cevONS15540XvrGEFC1310MTLC	cevContainerONS15540PortSlot	Gigabit Ethernet, fibre channel, 1310 nm SM, MTLC
cevONS15540XvrGEFC1310MTRJ	cevContainerONS15540PortSlot	Gigabit Ethernet, fibre channel, 1310 nm SM, MTRJ
cevONS15540XvrOC3Sonet1310	cevContainerONS15540PortSlot	SONET OC3, 1310 nm SM, MTRJ
cevONS15540XvrOC48Sonet1310	cevContainerONS15540PortSlot	SONET OC48, 1310 nm SM, MTRJ
Single-Port 10 Gb Ethernet Transponder (Two per LCMB 0401, 1400, and 1401 Series)		

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS1554010GigITUCh01	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 1
cevONS1554010GigITUCh02	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 2
cevONS1554010GigITUCh03	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 3
cevONS1554010GigITUCh04	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 4
cevONS1554010GigITUCh05	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 5
cevONS1554010GigITUCh06	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 6
cevONS1554010GigITUCh07	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 7
cevONS1554010GigITUCh08	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 8
cevONS1554010GigITUCh09	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 9
cevONS1554010GigITUCh10	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 10
cevONS1554010GigITUCh11	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 11
cevONS1554010GigITUCh12	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 12
cevONS1554010GigITUCh13	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 13
cevONS1554010GigITUCh14	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 14
cevONS1554010GigITUCh15	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 15
cevONS1554010GigITUCh16	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 16
cevONS1554010GigITUCh17	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 17
cevONS1554010GigITUCh18	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 18
cevONS1554010GigITUCh19	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 19
cevONS1554010GigITUCh20	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 20
cevONS1554010GigITUCh21	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 21

Table D-12 CTM-Supported ONS 15540 ESP Equipment (continued)

Equipment Name	Equipment Contained In	Description
cevONS1554010GigITUCh22	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 22
cevONS1554010GigITUCh23	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 23
cevONS1554010GigITUCh24	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 24
cevONS1554010GigITUCh25	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 25
cevONS1554010GigITUCh26	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 26
cevONS1554010GigITUCh27	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 27
cevONS1554010GigITUCh28	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 28
cevONS1554010GigITUCh29	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 29
cevONS1554010GigITUCh30	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 30
cevONS1554010GigITUCh31	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 31
cevONS1554010GigITUCh32	cevContainerONS15540SubCardSlot	10GE 1310 nm SM client with SC and ITU laser, channel 32

D.16 Equipment List for ONS 15540 ESPx

The ONS 15540 ESPx chassis supports all of the type I and type II transponder subcards supported by the ONS 15540 ESP. The transceiver modules for type II transponders are also supported for the ONS 15540 ESPx. All of the mux/demux subcards for the ONS 15540 ESP chassis are also supported.

The following table lists the additional ONS 15540 ESPx equipment that CTM R9.2 supports.

Table D-13 CTM-Supported ONS 15540 ESPx Equipment

Equipment Name	Equipment Contained In	Description
cevONS15540ESPxLCMBTspSp	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports four type I and II TSPs, with splitter
cevONS15540ESPxLCMBTsp	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports four type I and II TSPs, without splitter
cevONS15540ESPxLCMB10GOC192cSp	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports two 10GE/OC192c TSPs, with splitter
cevONS15540ESPxLCMB10GOC192c	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports two 10GE/OC192c TSPs, without splitter
cevChassisONS15540XPS	Chassis	—

Table D-13 CTM-Supported ONS 15540 ESPx Equipment (continued)

Equipment Name	Equipment Contained In	Description
ONS 15540 ESPx: Mux/Demux Modules Without OSC		
4 -Channel Module		
cevONS15540ESPxMDX4ChA	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band A, without OSC
cevONS15540ESPxMDX4ChB	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band B, without OSC
cevONS15540ESPxMDX4ChC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band C, without OSC
cevONS15540ESPxMDX4ChD	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band D, without OSC
cevONS15540ESPxMDX4ChE	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band E, without OSC
cevONS15540ESPxMDX4ChF	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band F, without OSC
cevONS15540ESPxMDX4ChG	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band G, without OSC
cevONS15540ESPxMDX4ChH	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band H, without OSC
8 -Channel Module		
cevONS15540ESPxMDX8ChAB	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band AB, without OSC
cevONS15540ESPxMDX8ChCD	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band CD, without OSC
cevONS15540ESPxMDX8ChEF	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band EF, without OSC
cevONS15540ESPxMDX8ChGH	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band GH, without OSC
16 -Channel Module		
cevONS15540ESPxMDX16ChEH	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 16-channel mux/demux, band EH, without OSC
Mux/Demux Motherboard		
cevONS15540ESPxMDXMBOsc	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports mux/demux modules, with OSC
cevONS15540ESPxMDXMB	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports mux/demux modules, without OSC
cevONS15540ESPxMDXMBOscESP	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports ESP mux/demux in ESPx chassis, with OSC
cevONS15540ESPxMDXMBESP	cevContainerONS15540CardSlot	ONS 15540 ESPx, supports ESP mux/demux in ESPx chassis, without OSC
ONS 15540 ESPx: Mux/Demux Modules with OSC		

Table D-13 CTM-Supported ONS 15540 ESPx Equipment (continued)

Equipment Name	Equipment Contained In	Description
4-Channel Module		
cevONS15540ESPxMDX4ChAOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band A, with OSC
cevONS15540ESPxMDX4ChBOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band B, with OSC
cevONS15540ESPxMDX4ChCOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band C, with OSC
cevONS15540ESPxMDX4ChDOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band D, with OSC
cevONS15540ESPxMDX4ChEOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band E, with OSC
cevONS15540ESPxMDX4ChFOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band F, with OSC
cevONS15540ESPxMDX4ChGOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band G, with OSC
cevONS15540ESPxMDX4ChHOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 4-channel mux/demux, band H, with OSC
8-Channel Module		
cevONS15540ESPxMDX8ChABOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band AB, with OSC
cevONS15540ESPxMDX8ChCDOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band CD, with OSC
cevONS15540ESPxMDX8ChEFOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band EF, with OSC
cevONS15540ESPxMDX8ChGHOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 8-channel mux/demux, band GH, with OSC
16-Channel Module		
cevONS15540ESPxMDX16ChADOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 16-channel mux/demux, band AD, with OSC
32-Channel Module		
cevONS15540ESPxMDX32ChOSC	cevContainerONS15540SubCardSlot	ONS 15540 ESPx, 32-channel mux/demux, with OSC

D.17 Equipment List for ONS 15600 SDH

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15600 SDH user documentation for actual equipment names used by the NE.

Table D-14 CTM-Supported ONS 15600 SDH Equipment

Card Category	Equipment
ASAP	ASAP_4
Controller	TSC
PPMs	PPM_1_PORT
	PIM_4_PPM
	PIM_1_PPM
STM cards	STM64_4
	STM16_16
	STM48_4
	STM64_4_DWDM
STM ports	STM16_PORT
	STM64_PORT
	STM1_PORT
	STM4_PORT
XC	SSXC
	CXC

D.18 Equipment List for ONS 15600 SONET

The equipment name is reported by the actual managed element. The following table lists the supported equipment names. For the most current list, refer to the Cisco ONS 15600 SONET user documentation for actual equipment names used by the NE.

Table D-15 CTM-Supported ONS 15600 SONET Equipment

Card Type	Equipment
ASAP	ASAP_4
Controller	TSC
OC cards	OC48_16
	OC48_32
	OC192_4
	OC192_4_DWDM
OC ports	OC3_PORT
	OC12_PORT
	OC48_PORT
	OC192_PORT

Table D-15 CTM-Supported ONS 15600 SONET Equipment (continued)

Card Type	Equipment
PPMs	PIM_4_PPM
	PPM_1_PORT
	PIM_1_PPM
XC	SSXC
	CXC

D.19 Equipment List for Unmanaged NEs

Unmanaged NEs do not report any equipment.

INDEX

A

accessing
- equipment inventory 3-55
- event channel 3-50
- subnetworks 3-123

adding
- managed elements 2-30
- multilayer subnetworks
  - drops 3-147
  - member subnetwork connections to VCATs 3-158

alarms
- disabling reporting of 3-76
- enabling reporting of 3-77
- retrieving
  - active 3-19
  - managed 3-82

all equipment names, retrieving 3-63

all supporting equipment, retrieving 3-72

all supporting equipment names, retrieving 3-72

attribute value change A-5

AVC events
- definition 2-29

C

cancelling rolls 3-167

common alarms A-10

common interface 3-18

completing rolls 3-168

configuring
- CTM GateWay/CORBA B-6

connecting to CTM 3-52

connection termination point names
- retrieving 3-94
- retrieving, in use 3-92

connection termination points
- naming conventions 2-31
- retrieving 3-92
  - in use 3-90

creating
- L2 topologies 3-195
- MLVLAN 3-180
- multilayer subnetworks
  - VCATs 3-156
- protection groups in a managed element 3-283
- rolls 3-163
- topological links
  - description 3-27
- VLANs 3-206

creating and activating
- multilayer subnetworks
  - subnetwork connections 3-146
  - subnetwork connections from user labels 3-138
  - subnetwork connections on DRI 3-172

CTM GateWay/CORBA
- fault management 1-3
- features 1-3
- information model 1-7
- overview 1-1
- services 1-4
  - naming service 1-4
  - notification service 1-4
  - overview 1-4

CTM GateWay/CORBA mapping

EMS A-13
MGX 8880/8850/8830 A-13
ONS 15216 A-24
ONS 15305 and ONS 15305 CTC A-26
ONS 15454 SONET R4.1 A-29
ONS 15530 A-50
ONS 15540 A-54
ONS 15600 SDH A-68
ONS 15600 SONET A-59
CTM GateWay/CORBA OSS users
logging out B-3
viewing B-2
CTM-specific details 2-23
interfaces 2-28
inventory 2-27
layer rate 2-23
naming conventions 2-30
object details 2-29

D

deactivating and deleting
multilayer subnetworks
subnetwork connections 3-149
deleting
L2 topologies 3-203
multilayer subnetworks
member subnetwork connections from VCATs 3-159
member subnetwork connections from VCATs, all 3-160
VCATs 3-158
PM data file 3-270
resources allocated for a session 3-288
VLANs 3-206
detecting a loss of communication 3-287
developing a CTM Gateway/CORBA client C-4
recommended tools C-4
disabling
alarm reports 3-76
performance monitoring 3-210
disconnecting a session 3-288
discovering protection groups for a managed element 3-277

E

edge termination points, requesting a list of 3-123
EMS
notification errors, informing NMS 3-208
public key, retrieving 3-53
public key pair, retrieving 3-54
session, accessing 3-52
session factory interface 3-52
session interface
description 3-50
event channels, accessing 3-50
EMS manager interface
adding
multilayer subnetworks 3-47
network partitions 3-47
software versions 3-50
creating
server trail links 3-43
topological links 3-27
deleting
multilayer subnetworks 3-49
server trail links 3-46
deleting and purging
managed elements 3-49
description 3-19
modifying topological links 3-32
retrieving
cleared and acknowledged EMS alarms 3-42
cleared and acknowledged EMS and ME alarms 3-41
L2 topologies 3-35
QoS template names 3-38

QoS templates, all 3-38
subnetwork names 3-23
system active alarms 3-20
topological link names 3-25
topological links, all top level 3-23
topological links, single 3-26
server trail links
creating 3-43
deleting 3-46
enabling
alarm reports 3-77
performance monitoring 3-211
equipment
holders
provisioning 3-75
unprovisioning 3-78
EquipmentHolder
naming conventions 2-35
equipment inventory manager interface 3-55
retrieving
all supporting equipment names 3-72
E-series VLAN interfaces 3-205
event channels, accessing 3-50
executing a protection switch 3-280

F

fault management, CTM GateWay/CORBA 1-3
files
history PM data 3-268
PM data 3-270
finishing
multilayer subnetworks
rolls 3-169
floating termination point, naming conventions 2-35

G

getCTMHistoryPMData
completed successfully 3-208
failed 3-208

H

heartbeat event A-11
history PM data, saving to a file 3-268

I

IDL-related, changes 1-8
information model 1-7
installation program B-12
interfaces 2-28
inventory 2-27

L

L2 topologies
creating 3-195
deleting 3-203
modifying 3-204
naming conventions 2-37
retrieving all top levels 3-35
L2 topology and MLVLAN for ML-series Ethernet cards 3-180
layer rate 2-23
location of naming service IOR files B-6

M

maintenance commands
finding 3-79
releasing 3-80
maintenance manager interface 3-79
managed element manager interface 3-82

managed element names
requesting
information about 3-87
managed elements
creating protection groups in 3-283
deleting protection groups from 3-286
naming conventions 2-30
requesting
information about 3-85
retrieving 3-95
managers, accessing 3-51
managing
client-server connection 3-287
protection 3-276
MDS 9000 2-22
naming conventions D-1
MGX 8880/8850/8830 2-22
naming conventions D-2
MLVLAN
adding front ports to 3-183
creating 3-180
deleting 3-189
disabling traffic 3-185
dropping front ports from 3-184
enabling or disabling traffic on drop ports 3-185, 3-186
enabling traffic 3-185
modifying
customer ID 3-186, 3-188
service ID 3-186, 3-188
naming conventions 2-37
retrieving
all MLVLAN names 3-191
information on 3-189
list of all 3-190
modifying
L2 topologies 3-204
topological links, description 3-32
monitoring performance 3-210
multilayer subnetworks
cancelling rolls 3-167
completing rolls 3-168
definition 2-29
deleting rolls 3-166
finishing rolls 3-169
forcing valid signals 3-167
multilayer subnetwork interface 3-123
naming conventions 2-30

N

naming conventions 2-30
connection termination point 2-31
EquipmentHolder 2-35
floating termination point 2-35
L2 topology 2-37
ManagedElement 2-30
MDS 9000 D-1
MGX 8880/8850/8830 D-2
MLVlan 2-37
MultiLayerSubnetwork 2-30
ONS 15216 D-2
ONS 15305 D-4
ONS 15305 CTC D-4
ONS 15310 CL D-5
ONS 15310 MA SONET D-5
ONS 15327 D-6
ONS 15454 SDH D-6
ONS 15454 SONET D-9
ONS 15530 D-13
ONS 15540 D-26
ONS 15600 SDH D-34
ONS 15600 SONET D-35
physical termination point 2-30
protection groups 2-36
published CTM GateWay/CORBA objects B-5
QoSTemplate 2-37
topological links 2-36
unmanaged NEs D-36

VCAT 2-37
naming services
description 1-4
on CTM server 1-5
on independent servers 1-7
on NMS workstation 1-6
NE-specific details 2-1
MDS 9000 2-22
MGX 8880/8850/8830 2-22
ONS 15216 2-1
ONS 15305 and ONS 15305 CTC 2-2
ONS 15310 CL 2-2
ONS 15310 MA 2-6
ONS 15327 2-7
ONS 15454 SDH 2-9
ONS 15454 SONET 2-12
ONS 15530 2-14
ONS 15540 2-16
ONS 15600 SDH 2-21
ONS 15600 SONET 2-18
unmanaged NE 2-22
nmsSession
NmsSession_I 3-208
NMS session interface 3-208
notification errors, informing NMS 3-208
notification services
description 1-4
on CTM server 1-5
on independent servers 1-7
on NMS workstation 1-6

O

object creation A-3
object creation event 2-30
object deletion A-4
object details 2-29
ONS 15216 2-1
naming conventions D-2
ONS 15305
naming conventions D-4
ONS 15305 and ONS 15305 CTC 2-2
ONS 15305 CTC
naming conventions D-4
ONS 15310 CL 2-2
naming conventions D-5
ONS 15310 MA 2-6
ONS 15310 MA SONET
naming conventions D-5
ONS 15327 2-7
naming conventions D-6
ONS 15454 SDH 2-9
L2 service interfaces 3-180
naming conventions D-6
ONS 15454 SONET 2-12
L2 service interfaces 3-180
naming conventions D-9
ONS 15530 2-14
naming conventions D-13
ONS 15540 2-16
naming conventions D-26
ONS 15600 SDH 2-21
naming conventions D-34
ONS 15600 SONET 2-18
naming conventions D-35
optical monitoring alarms
ONS 15530 and ONS 15540 A-57
OSS client profile for CTM GateWay/CORBA
creating B-1
deleting B-2
overview
CTM GateWay/CORBA 1-1
overview of notifications A-1

P

parameters, not supported 1-9
performance management interface 3-210

performance monitoring
disabling 3-210
enabling 3-211
retrieving threshold values on TPs or layer rate measurement points 3-267
physical termination point names, requesting all 3-89
physical termination points
naming conventions 2-30
requesting all 3-88
retrieving
all supported PTP names by specific equipment 3-71
that can carry protected traffic 3-277
used as subnetwork connection source 3-102
using to retrieve topological links 3-99
ping command 3-288
PM data
deleting file 3-270
requesting
all current 3-274
filtered 3-271
number of hours stored in CTM 3-274
probable cause mapping A-12
programming details 2-37
protection groups
creating in a managed element 3-283
discovering for a managed element 3-277
naming conventions 2-36
retrieving associated protection groups 3-287
retrieving the current status of 3-279
supported for the protection command (table) 3-281
protection manager interface 3-276
protection switch A-8
executing 3-280
provisioning
equipment holders 3-75
subnetwork connections 2-39

Q

QoS
templates
names, retrieving all top levels 3-38
naming conventions 2-37
retrieving all top levels 3-38
retrieving information 3-41

R

releasing maintenance commands 3-80
requesting
all managed element names 3-126
list of edge termination points 3-123
managed element
information 3-85
names, by associated subnetwork 3-126
physical termination point names 3-89
physical termination points 3-88
PM data
all current 3-274
filtered 3-271
subnetwork connections
by managed element 3-83
resolving
naming service-related issues using debugging utilities B-6
notification services A-2
resynchronizing the NMS from the EMS
using cases C-1
retrieving
all supporting equipment names 3-72
associated protection groups 3-287
capabilities of the common interface 3-18
connection termination point names 3-94
connection termination point names in use 3-92
connection termination points 3-92
in use 3-90

to use as a subnetwork connection source 3-103
CTPs and topological links 3-134
EMS
information 3-33
equipment 3-74
equipment holders 3-73
L2 topologies
associated with an MLVLAN 3-194
associated with a subnetwork connection 3-193
description 3-35
structure 3-193
Layer 1 SONET/SDH subnetwork connections, all 3-192
list of VLANs 3-207
managed alarms 3-82
managed elements 3-95
ML-series equipment names (for RPR L2 topology) 3-204
multilayer subnetworks 3-133
associated roll names for subnetwork connections, all 3-171
associated subnetwork connection names for roll, all 3-171
roll names, all 3-170
rolls, all 3-169
subnetwork connections for merge 3-161
topological links 3-137
VCAT member subnetwork connection names, all 3-155
VCAT member subnetwork connections, all 3-154
VCAT subnetwork connection names, all 3-153
VCAT subnetwork connections 3-150
physical termination points
that can carry protected traffic 3-277
to use as a subnetwork connection source 3-102
QoS template names, description 3-38
QoS templates
all 3-38
subnetwork connection names with TP, all 3-130
subnetwork connections by user label 3-136
subnetwork connection structure 3-135
subnetwork connections with termination points 3-128
subnetwork names
description 3-23
system active alarms 3-20
TCA termination point PM parameters 3-267
topological link names, description 3-25
topological links
all 3-23
single 3-26
using the PTP name 3-99
running CTM GateWay/CORBA clients C-11

S

sample code in Java C-5
connecting StructuredPushConsumerImpl C-11
getEMS operation on EMS manager C-9
implementing
NmsSession_IOperations C-7
StructuredPushConsumer C-10
initializing client connections C-5
logging in and retrieving EmsSession C-8
obtaining
ConsumerAdmin references C-9
ProxyPushSupplier C-10
retrieving
EMSSessionFactory references C-7
EventChannel references C-9
list of managers C-9
naming service references C-7
saving history PM data to a file 3-268
session
deleting allocated resources of 3-288
disconnecting 3-288
interface 3-287
setting

termination point parameters 3-100
software and data manager interface 3-295
starting and stopping CTM GateWay/CORBA services
using CLI B-8
state change event A-6
subnetwork connections
CTPs that can be used as a source 3-103
PTPs
used as a source 3-102
requesting
descriptions 3-83
retrieving, all 3-127
supported parameters (table) 3-143
support for 2-29
subnetwork connections names
retrieving, all 3-130
subnetwork names, retrieving all top level 3-23
subnetworks
retrieving
list of edge termination point names 3-124
names of 3-94
topological link names, all 3-132
topological links, all 3-131
supported events A-3
attribute value change A-5
common alarms A-10
heartbeat event A-11
object creation A-3
object deletion A-4
protection switch A-8
state change event A-6
threshold crossing alert A-9
supported managers, accessing 3-52
system alarms, retrieving 3-20

T

TCAs
modifying thresholds of 3-212
termination points
retrieving structure of 3-98
setting parameters for 3-100
threshold crossing alert A-9
tips
improving notification throughput A-2
TMF-defined probable cause A-12
TMF interfaces
common 3-18
EMS manager 3-19
EMS session factory 3-52
equipment inventory manager 3-55
maintenance manager 3-79
managed element manager 3-82
multilayer subnetwork 3-123
NMS session 3-208
performance management 3-210
protection manager 3-276
session 3-287
using 3-1
topological link names, retrieving all top level 3-25
topological links
creating 3-27
deleting 3-31
modifying 3-32
naming conventions 2-36
retrieving
all top level 3-23
top level 3-26
using the PTP name 3-99
with Y-cable protection, creating 3-29

U

unmanaged NE 2-22
unmanaged NEs
naming conventions D-36
unprovisioning equipment 3-78
upgrading

multilayer subnetworks 3-162

using

encryption between OSS client and CTM Gateway/CORBA B-3

multiple naming servers B-4

V

VCAT

naming conventions 2-37

VLANs

creating 3-206

deleting 3-206

retrieving list of 3-207